

;login:

March/April 1994 Vol. 19, No. 2

Growing Up in the 90's

I have a confession to make. For some reason, I appear to have lost a certain measure of respect and awe I had for hardware technology. For years and years, each new improvement (e.g., bigger RAMs, faster CPUs, faster displays, reduced power consumption, reduced cost) amazed and astounded me.

Regrettably, I have come to expect it. I have been working in the world of commodity hardware (i.e., *gulp* PCs) and now have come to believe that each month I'll see reduced cost for disks, faster CPUs, and, in a terribly selfish way, some set of better peripherals to appeal to me.

This is a terrible situation! It ignores the complexities of creating hardware (much less software to drive the hardware), the incredible difficulties of creating circuit designs, testing, distribution, etc. etc. I talk to people about 25 MIPS PCs, 80 MIPS SPARC engines, and 100 MFLOPs PowerPCs as if they are commodities that have been available forever. The hardware promises made 10 years truly are coming to fruition. On schedule. And under budget.

I even expect to find a whizzy new computer game every month or two. *SimCity 2000* was my latest acquisition -- it's a full measure better than *SimCity Classic*. Its user interface is much better and the complexity of the game seems just about right. I haven't tried *CAESAR* yet and still haven't mastered *Civilization* (the Mongolian Hordes get me every time).

Why is this irksome to me? Because I'm in the business of awing people with technology. As I become jaded, I imagine the consumer base is much more so.

What a pity, too; technology and complexity are so neat. I hope I can regain at least some of my innocence. RK

The closing date for submissions to the next issue of ;login: is April 15, 1994.

Association News

Winter '94 Conference Reports..... 3

Board Meeting Summary..... 10

USL and UC Berkeley Reach Agreement 12

Lifetime Achievement Awards 13

Computing Systems Editorship 13

Community News..... 13

USENIX Member Benefits..... 14

Vendor Exhibitions..... 14

SAGE News

Living with CC:Mail 15

Hal Pomeranz

SAGE Views 17

Elizabeth Zwicky

Perl Practicum 18

Hal Pomeranz

Features

PLATO: The Emergence of On-Line Community 20

David R. Woolley

The God of Backup 28

Greg Rose

What's New 29

News from EurOpen National Groups..... 33

Alexandru Rotaru

CitySpace..... 43

;login: 50 and 100 Years Ago 49

Barry Shein

An Update on UNIX-related Standards Activities 34

Nicholas Stoughton

Book Reviews

The Bookworm 44

Peter H. Salus

A Second Look..... 45

Frank Hecker

Applied Cryptography..... 46

Donald T. Davis

TCP/IP Illustrated 47

Vern Paxson

Sendmail..... 48

David J. Fiander

Announcements/Call for Upcoming Events 49

Book Order Forms..... 65

Local User Groups 72

Calendar of Events..... 75



General Information

login: is the official newsletter of the USENIX Association.

login: (ISSN 1044-6397) Volume 19, Number 2 (March/April 1994) is published bimonthly by the USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710. \$24 of each member's annual dues is for an annual subscription to *login*. Subscriptions for nonmembers are \$50 per year. Second-class postage paid at Berkeley, CA and additional offices. POSTMASTER: Send address changes to *login*, USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710.

Contributions Solicited

You are encouraged to contribute articles, book reviews, and announcements to *login*. Send them via email to <login@usenix.org> or through the postal system to the Association office. Send SAGE material to <bigmac@erg.sri.com>. The Association reserves the right to edit submitted material. Any reproduction of this newsletter in its entirety or in part requires the permission of the Association and the author(s).

Editorial Staff

Rob Kolstad, Editor <kolstad@usenix.org>
Bryan McDonald, SAGE Editor
<bigmac@erg.sri.com>
Nick Stoughton, Standards Report Editor
<nick@usenix.org>
Carolyn S. Carr, Managing Editor & Typesetter
<carolyn@usenix.org>
Monica Jensen, Copy Editor
Lilia Scott, Proofreader

Membership and Publications

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710
Telephone: 510/528-8649
FAX: 510/548-5738
Email: <office@usenix.org>

Copyright © 1994 USENIX® Association. The Association acknowledges all trade references made herein.

login: is produced with Framemaker 3.1 software provided by Frame Technology and displayed on an NCD X Terminal, donated by Network Computing Devices. The system is served by a Sun SPARCsystem 10 server. Output is sent to a 600 dpi QMS 860 laser printer, donated by Quality Micro Systems. The newsletter is printed on recycled paper. ♻

Conferences & Symposia

Judith F. DesHarnais, Conference Coordinator
USENIX Conference Office
22672 Lambert Street, Suite 613
Lake Forest, CA 92630
Telephone: 714/588-8649
FAX: 714/588-9706
Email: <conference@usenix.org>

Automatic Information Server

To receive information electronically about upcoming USENIX symposia & conferences, finger <info@usenix.org> and you will be directed to the catalog which outlines all available information about USENIX services.

Tutorials

Daniel V. Klein, Tutorial Coordinator
Telephone: 412/421-2332
Email: <dvk@usenix.org>

USENIX Supporting Members

ANDATACO
ASANTÉ Technologies, Inc.
Frame Technology Corporation
Matsushita Electrical Industrial Co., Ltd.
Network Computing Devices, Inc.
OTA Limited Partnership
QMS, Inc.
Sybase, Inc.
UUNET Technologies, Inc.

SAGE Supporting Member

Enterprise Systems Management Corporation

USENIX Association Board of Directors

If a member wishes to communicate directly with the USENIX Board of Directors, he or she may do so by writing to <board@usenix.org>.

President:

Stephen C. Johnson <scj@usenix.org>

Vice President:

Michael D. O'Dell <mo@usenix.org>

Secretary:

Evi Nemeth <evi@usenix.org>

Treasurer:

Rick Adams <rick@usenix.org>

Directors:

Eric Allman <eric@usenix.org>

Tom Christiansen <tchrist@usenix.org>

Lori Grob <grob@usenix.org>

Barry Shein <bzs@usenix.org>

USENIX Association

Executive Director

Ellie Young <ellie@usenix.org>

Vinter '94 Conference Reports

Best Paper/Presentation Awards

The Best Paper award was presented to Udi Manber and Sun Wu, University of Arizona, for "GLIMPSE: A Tool to Search Through Entire File Systems."

J. Bradley Chen from Carnegie Mellon School of Computer Science won Best Student Paper, "Memory Behavior for an X11 Window System."

The Best Presentation award was given to Rob Pike, AT&T Bell Laboratories, for "Acme: A User Interface for Programmers."

Invited Talks

Reviewed by Peter Collinson

<pc@hillside.co.uk>

UNIX on Wall Street

Marc Donner

Marc is now a VP at Morgan Stanley, an investment bank based in a tower near Radio City Music Hall in New York City. His talk was in three parts, the first described the financial services industry, the second covered why financial services firms use so much computing, and the third included visions of things that might make running systems easier.

Wall Street is split into two camps: people who buy and sell securities and people who help others issue securities. There are two main types of securities, though the distinctions are blurred: equities, otherwise known as stocks, and fixed income, once called bonds.

Trading floors are the places where the business is done. Traders often have leased lines to their counterparts in other banks so that they can contact someone to make a deal instantly. Price information comes from market data sources. Each trader deals with a small number of "instruments" or types of security. Each trader has an "inventory" of his or her product, which is called a book. During the trading day, trades are written down on a sheet called a "blotter" and also written on trade 'tickets' which are passed to the back office for recording and execution. The operation is heavily regulated.

The main problem is estimating and predicting prices. This demands a huge amount of computing.

Marc has been responsible for building part of Morgan Stanley's world wide network. The big concern is that machines should never go down. Failure can be very expensive. Users don't care why a machine is down, nor should they. Morgan Stanley stocks spares and plugs in a new machine if one breaks. Marc is not happy with the maintenance services offered by hardware vendors. Prices run five times the cost of time and materials charges when averaged over hundreds of machines, and the service standard is low.

Marc concluded with WIBNI's - "Wouldn't It Be Nice If":

- Wouldn't it be nice if heterogeneity worked in practice?
- Wouldn't it be nice if you could add extra display screens without serious system alteration?
- Wouldn't it be nice if you could manage desktop machines from the network, even if the operating system had crashed.

The Facts About Fax

Ed McCreigh, Adobe Systems Inc

This talk was one of the highlights of the conference for me (and for others). It covered the history of Fax transmission, the mechanics of modern systems and some idea of future direction.

Fax transmission has been used for 150 years; pictures were sent over telegraph lines. The first commercial Fax machine was made in 1861; the next steps were between 1900 and 1909 using optical printing systems and film.

AT&T's system for newspapers to send pictures began in 1924. A big breakthrough to remove AT&T's monopoly came in 1935. In 1965, the Magnavox-Xerox telecopier was introduced. Its protocol was the first Group 1 Fax. The CCITT Group 1 Fax standard was brought out in the same year. The U.S. deviated from this. The Group 1 standard provided a transmission speed of 6 mins/page and required much operator attention.

The first real-world standard was CCITT Group 2 Fax unveiled in 1976. It permits transmission speeds of 3 mins/page. There are still some Group 2 Fax machines in use. The boom in Fax machines began with the Group 2 standard but

has been fuelled by the Group 3 standard (1980). This standard enables a transmission speed of about 30 secs/page. There is also a Group 4 standard that was approved in 1984, but it's not used widely yet.

Worldwide, there are now about 40 million group 3 fax machines. The growth is 30%/year. Prices are falling. Almost all the machines are of Japanese manufacture.

Computer fax transmission is the next step. Frecom YM7109, the first computer modem, was supplanted by modems called "Class 1" and "Class 2" fax modems. Class 2 modems are pre-loaded with the critical information and operate their own protocol. The computer does much less and is not required to make decisions under certain time constraints. Class 2 modems are less flexible than Class 1 modems because the protocol usage is defined by the modem manufacturer, not by software. Class 4 modems are new and are designed to permit Class 1 flexibility by complete pre-loading of the time-dependent stuff.

Ed's talk finished with some idea of the future directions for fax. He expects to see faster transmission, using V17 modems, ISDN; some resolution of the problem of personal fax addressing; more compression methods; more file transfer methods being embedded in fax and more image representations being used; and work on security. Apparently, it is a standard practice at one U.S. bank to call the sender of a fax and ask for a retransmission quoting a certain ID number. This way the bank is more confident that the fax is not a forgery.

Objecting to Objects

Steve Johnson, Melismatic Software

The President of the USENIX Association, Steve Johnson, devoted his talk session to a discussion of Object Oriented Programming, OOP. Steve's view is that this subject is greatly hyped at present, and the benefits to the programmer and the user community are overated.

Steve opened by saying that Objects is a good thing, but it is merely a step. Unfortunately, hype has stopped the evolution of Objects. The hype notes that objects are good because they focus on software models; they encapsulate experience in a re-usable way; and they permit fine control of interfaces. However, none of these reasons are anything to do with OOP-ness.

Steve defines OOP to be a programming model supporting classes with inheritance, objects with behavior determined by class, objects communication by messages or methods, data hiding, and

protection. He does not assume separate threads for objects or for polymorphism. In reality, he is thinking about C++.

Steve has several technical objections to OOP. First, OOP is not a new idea: it dates to Simula (1967), which was not too successful because it cost too many machine cycles. Steve's concern is that we are wasting cycles now because they seem plentiful. He is a 'cycle-miser'.

The second technical objections is that OOP claims simply are not met. There is no proof that it is quicker to develop things in OOP.

One argument is that OOP facilitates code reuse. This is not an OOP feature. For code to be reusable, it must share a common model of operation. Reuse implies relevant encapsulated knowledge, and the knowledge must be needed, in that form, by the reuser.

Another OOP claim is higher reliability for OOP code. The checking of function arguments is an advance, but Pascal and Ansi C do it too (as did Algol). However, messages are frequently unchecked: "Messages are the GOTO's of the 90's."

OOP makes programmers focus on style and not content. The quality of code depends largely on the quality of design, the depth and relevance of the programmer's knowledge, and the care taken in construction and testing. OOP is not a substitute for thought.

Steve examined the 'social' aspects of OOP as well. First, is OOP used to address the correct problems?

He concluded this section with a discussion of Persistent Objects: he is a fan of them but feels that support for Persistent Objects is too rudimentary at present.

The talk ended with Steve's re-iterating his main point: OOP is a fad and is over-hyped. While it should be a consideration, it should not be the ultimate way of doing things. The overall mark is something like C-, and the teacher's comment: "Could do better."

Invited Talks

Reviewed by Bob Gray

[<rgray@advtech.uswest.com>](mailto:rgray@advtech.uswest.com)

Beyond LEX and YACC: How to Generate the Whole Compiler

William. M. Waite, University of Colorado

Bill Waite gave an animated and informative talk on how to build fast, efficient compilers from specifications. His entire talk was based on a spe-

cific example: given a set of simple line-oriented "set" specifications, generate a bunch of "C" header data structures. This translation task requires checking name uniqueness, avoiding set replication, and counting set names and members. Bill explained that the parsing and scanning specifications are pretty-well understood but we need tools to help with the rest. He told us about "PTG," a tool to output fragments of an abstract tree using "printf"-like patterns. The specifications for the example at hand are short and transparent. Bill explained a bit about automatic tree-walking and attribute computation. He showed us that a tool could carry-out much of the work involved with these tasks. Finally Bill talked about the advantage of integrating compiler tools into a coherent system – that specification redundancy could be eliminated. The whole compiler system is available via ftp at <ftp://ftp.cs.colorado.edu/pub/distrib/eli> and a detailed example paper has been published in the Invited Talks Notes.

Video Compression — What To Do When Everything Is Changing?

Lawrence A. Rowe, EECS, University of California

Larry gave a terrific talk on Video Compression – how the compression algorithms work, what the state of the art is now, and what hardware exists. He reviewed the extremely high data rates needed for uncompressed digital video, especially for High Definition TV (up to 78 MBytes/sec). We learned a bit about line sampling and down sampling before we saw the basics of spatial encoding, the block transform encoding. This is used for JPEG, MPEG and H.261.

The Discrete Cosine Transform better represents video information than, for example, the Fast Fourier Transform. We learned that images can be encoded with about 0.55 bits/pixel. Larry doesn't believe that there will be much activity for other compression algorithms such as DVI, CellB, Fractals or Wavelets, at least in the near future. MPEG will be the most-used compression algorithm, partly because it is non-proprietary. We learned that MPEG yields decent video from 1.5 Mbps streams, and its pretty good at 3-5 Mbit/sec. Larry introduced some of the jargon, I, P and B frames and motion vectors, and explained how they make MPEG different from JPEG. He talked about MPEG performance: decoding is relatively easy and today's workstations are almost powerful enough for real-time display. However, the encoding is still very expensive. Finally, Larry talked about the Berkeley Plateau Project – a Video-On-Demand system that will soon be available to the public

Refereed Sessions

Reviewed by Bryan Costales

<bcx@icsi.berkeley.edu>

Finding Similar Files in a Large File System

Udi Manber, Dept. of Computer Science, University of Arizona

Files are similar when they contain a significant number of identical sections of text. The author described his "sif" program as a method to detect similarities among many files in a large file system. It is fast, processing approximately 1/2 to 1 Gb of text per hour (with I/O the limiting factor), but it only works with text files (as opposed to binary or compressed files). Conceptually adapted from biology, "sif" uses a "fingerprint" inspired by "enzyme tags." Run in two passes, a database (typically 5% the size of the searched data) is first built to index fingerprints (4-byte checksums of 50-byte chunks of text) into files. Later search-runs use the index to quickly find similarities based on the number of identical fingerprints in common. The threshold of similarity is tunable. Output consists of lists of file index numbers and fingerprint numbers that are in common. The output can seem confusing, but can be statistically reduced to more informative forms. Fortunately such reduction will be a part of future releases of "sif." Unfortunately, "sif" is not yet available.

cql – A Flat File Database Query Language

Glenn Fowler, AT&T Bell Laboratories

The author recognized that a gray area exists between line-oriented tools like "awk" and "sed" on the one hand, and relational databases on the other. The result was "cql", a C-like query language used to search flat files like */etc/passwd*. It is a query language in that it only searches and does not update (write to) files. The author compared "cql" to other common tools like "perl." "cql" is special purpose, so tends to be faster than general purpose programs; "cql" is designed to be a database interface, so can easily handle multiple schema and automatic closure; and "cql" uses C-language like-expressions, thereby reducing learning time. "cql" expressions can be read from files, so libraries of expressions can be used to build complex systems. Regrettably, "cql" is not yet available.

GLIMPSE: A Tool to Search Through Entire File Systems

Udi Manber, Dept. of Computer Science, University of Arizona

In his second appearance of the day, the author presented his winner-of-the "Best Paper Award." GLIMPSE (an extension of the author's 1992

"agrep" program) stands for GLoBal IMplicit SEarch. GLIMPSE allows (via indexes) files throughout the entire filesystem to be quickly searched for approximate matches (ala "agrep"). For example, "agrep" with 2 errors matches the words "juvenile" and "senile" in */usr/dict/web2*. GLIMPSE took 5 seconds to match 19 occurrences of the words USENIX and Winter in 4300 files. GLIMPSE features a small index (2 to 4% of the size of the data), flexible searches (similar to "agrep"), approximate matches, and easy customizations. On the down side, it is less fast than it perhaps should be and is limited to file <500Mb in size. GLIMPSE gracefully excludes non-ASCII and other problem files like uuencoded ones. The author admitted that GLIMPSE is not good for biological files. DNA files, for example, bloat the index to 10%. Future enhancements to GLIMPSE may be context-based searches and searches for compressed files. This software will be available "very soon now."

Refereed Sessions

Reviewed by Bryan Costales

<bcx@icsi.berkeley.edu>

A Uniform Name Service for Spring's UNIX Environment

Authors: Michael N. Nelson, Silicon Graphics, and Sanjay R. Radia, SunSoft, Inc.

Presenter: Sanjay R. Radia

Names in UNIX are of mixed types. In directories, for example, a string is looked up, and the integer value of an inode returned. In */etc/printcap*, on the other hand, a string is looked up and a string returned. Under the Spring operating system, all name spaces are objects. The philosophy of Spring is that all objects (including things like shell variables, file names, and devices) are accessed via common methods.

Here I must digress. Throughout this talk, use of the word "name" was confusing. A better choice would have been "identifier." Similar confusion was caused by the use of "domain" to mean "process," and "village" to mean LAN. This, combined with an absence of examples, left all but a small handful of the audience completely lost.

The claim was made that Spring is superior to Plan 9 in many ways. This was disputed during the question period.

ACID: A Debugger Built from a Language **Phil Winterbottom, AT & T Bell Laboratories**

ACID is an interpreted language used to debug programs by running them. The ACID language is patterned after C, but is a scripting language rather than a compiled one. ACID scripts can be produced by running specially compiled pro-

grams. ACID scripts can also be built from primitives (like the internal `pcfile()` function) to create complex debuggers customized to particular programs or needs. An example was given of a memory leak-detector written with 120 lines of ACID code. When asked if ACID is available, the author replied, "If you are a Plan 9 licensee, it is on the disk."

Acme: A User Interface for Programmers

Rob Pike, AT&T Bell Laboratories

Acme is an X11 window manager specifically designed to manipulate text. In it, the left mouse button is "normal." The middle button causes the text under the cursor to be executed as either a built-in command, a program in the current directory, or a program from a central bin. The right button selects the current context. For example, a file name will cause that file to be loaded (or the cursor to be moved to that window if it is already loaded). Under Acme, the windows display in columns with header stripes visible for selection. Windows are automatically created as context progresses and destroyed in the reverse progression. Acme understands compiler errors, so clicking on a line of error output will cause the referenced source file to be loaded, the offending line displayed, and the cursor moved to that line in the new window. The author admitted that this approach takes "some getting used to." He attempted to demonstrate Acme on a laptop connected to the video projector, but the "demo effect" kicked in and many tasks failed; his lesson, always bring a prepared video. Like ACID, Acme is only available to Plan 9 licensees.

File System Design for an NFS File Server Appliance

Authors: Dave Hitz, James Lau, and Michael Malcolm, Network Appliance Corporation

Presenter: Dave Hitz

The author described WAFL (a Write Anywhere File Layout file system) which is in use on the NFS-toaster, a product of Network Appliance. He stressed that this is a "real" filesystem. It has been in use at Network Appliance for a year and a half, and has been in actual use on 100+ units sold over the last six months. The author then described WAFL in detail: A root inode that points to an inode "file" is key to WAFL and one of two ways it differs from the Berkeley file system. The second departure is that writes can commit to disk anywhere the head happens to be. These differences combine to make WAFL faster and more reliable. Reliability is achieved through Snapshots. The disk cache is flushed and the root inode copied (a snapshot taken) once every ten seconds. Crash recovery is swift because the current root inode is abandoned in favor of the most recent snapshot. Almost as a side effect, snapshots enable disk backups that are

automatic, tunable, and incremental. Users can recover from snapshots using ordinary utilities like *cp*.

Improving the Write performance of an NFS Server **Chet Juszczak, Digital Equipment Corporation**

The author first described NFS, the NFS protocols, and the roles of *nfsd* and *biod*. Next, he outlined his attempts to improve NFS write performance at the kernel and *nfsd* levels. After describing several approaches ("write gathering" and "procrastination"), he summarized. In general, any scheme that can avoid disk writes at the NFS level is a "big win." But beware, NFS is inherently tied to the underlying filesystem structure, and no one scheme is suitable to all. Further work in this area is needed.

Not Quite NFS, Soft Cache Consistency for NFS **Rick Macklem, Dept. of Computing and Information Science, University of Guelph**

This talk described NQNFS (Not Quite NFS), an experimental modification of NFS to improve write performance on the client-side. The author attempted to "decouple the client from the server with client-side caching." Key to this approach is trading elimination of write-through on close, with a "callback" mechanism to maintain cache consistency. The callback is implemented with "leases" where the client maintains the server's state for windows of time. The author concluded that this approach is beneficial, but further work is still required.

Refereed Session on Wide-Area Information Access

Reviewed by Win Bent

<whb@skat.usc.edu>

Drinking From the Firehose: Multicast USENET News **Kurt J. Lidl, Josh Osborne, Joe Malcolm, UUNET Technologies, Inc.**

The Mbone, the Internet multicast backbone, has been the focus of a fair amount of attention recently, particularly in the realm of multimedia. It was announced that the Keynote Address was being broadcast over the Mbone, and that it had been hoped to make that a two-way connection. This paper focused on a more traditional use, the ever-more-popular USENET. Using this system, news articles which might have gone over the same net many times get broadcast once, which reduces net traffic and shortens propagation delay. Despite the Mbone's known unreliability (in other words, you might not get all the articles), the authors have made their system very trustworthy (in other words, when you do get an article, you know it's real).

The redbms Distributed Bibliographic Database System

Richard A. Golding, Vrije Universiteit, Darrell D.E. Long, University of California, Santa Cruz, and John Wilkes, Hewlett Packard Laboratories

This talk described "a system for sharing bibliographic references among many users," to quote the abstract. Using a format slightly modified from Lesk's *refer*, the software provides tools for adding, editing, sending, and receiving bibliographic entries, as well as searching one or many databases scattered throughout the net. My view of this system is that a good first pass has been made at a potentially huge task, setting up a framework which could well become a standard.

File System Daemons as a Unifying Mechanism for Network Information Access

Steve Summit, Consultant, Seattle, Washington

I really like the impetus for this talk's software: the author wanted to use *ls* and *grep* on data obtainable through *gopher* and *WWW*, rather than remain tied to the single interface previously available. Unfortunately, his solution has a few flaws, which were pointed out by the audience. Upon hearing that I'd have to recompile and/or relink all the tools which expect to use this scheme, I gave a silent "Eeewww..." More importantly, many seemed puzzled – is this a new idea? How is it different from, for example, *Prospero*? Still, despite this, I like the concept.

Steve Summit, the author of this last paper, one was genuinely interesting to listen to! The first two speakers were too low-key to generate or maintain enthusiasm, despite their very useful ideas. They seemed to say, "If the idea sells itself, then I'll just read the proceedings." I like to attend talks in which the author is upbeat and lively.

Refereed Session on Dealing With The Pc World

Reviewed by Win Brent

<whb@skat.usc.edu>

Wux: UNIX Tools under Windows **Diomidis Spinellis, Dept. of Computing, Imperial College of Science, Technology, and Medicine**

This talk described a library which allows UNIX tools to be recompiled to run under Windows without modification, thus making the tool think it's still dealing with *stdin* and *stdout* when it's actually dealing with windows! Pretty sneaky, huh? This is the type of problem which gets worse the closer you look, and Diomidis Spinellis looked very closely. Although the software is not

currently available, Spinellis hinted that it would be, once he had made it more presentable.

An Overview of the NetWare Operating System
Greg Minshall, Drew Major, Kyle Powell, Novell, Inc.

This talk, a breathlessly fast look at the guts of NetWare, showed that there's more to client-server computing than adding networking to and existing OS. One of the main selling points of NetWare is that it is usable by a variety of computers and OSs, which means that the mid- and lower-level routines have to be fairly generic. In addition, the author showed how NetWare has been tuned to provide quick response for a specific type of workload: frequent, short-duration requests. He made some interesting points, although I would like to have seen some comparisons to similar functions in UNIX - this WAS a UNIX conference, wasn't it?

The USENIX Winter 1994 Contest:

Design An Industry Alliance

by Ed Gould

<ed@pa.dec.com>

Following occasional tradition, we again held a competition at the Winter 1994 Conference. This time, the goal was to suggest a computer industry merger or alliance, and name the product it would sell. For example:

- MicroSoft and Disneyland collaborate on "Pirates of Puget Sound"
- USL and Coco Chanel standardize on "Chanel No. 5 Release 3.2"
- DEC and Crisco co-develop "Cholesterol-free Snake Oil"
- Novell and Christian Dior sell "ReadyToWare"

The winners were, in order of presentation — last to first:

Honorable Mentions:

Best Disqualified

Dan Klein <dvk@lonewolf.com>
DEC, HP, IBM, Siemens,...., form an alliance called OSF (oh, sorry, it's been done...)

Best Ergonomics

Irving Reid <irving@platform.com>
Digital Equipment Corp. and LA-Z-Boy co-produce the DECchair

Best GNU Joke

James Buszard-Welcher <jwelcher@ssf-sys.dhl.com>
GNU and McDonalds sell EMC's

Best Guarantee

Angela Thomas <angela@tandem.com>
Domino's Pizza and FSF guarantee a new release in 30 minutes or less, or your money back

Best Graphically

Dan Klein <dvk@lonewolf.com>
Violent Alliance between Apple and Smith & Wesson to produce the "Point and Shoot" User Interface

Best Food

Craig Leres <leres@ee.lbl.gov>
AT&T and Kellogg's sell a new breakfast cereal called "Dark Fibre"

Best Attempt to Bribe a Judge

Jim Duncan <jim@math.psu.edu>
An alliance between: Palo Alto Startup, USENIX Board, Greg Rose, and NTP developers produces: ParcPlace, SAGE, RoSecure, and Time

The Winners:

4th Place

Ben Fried <ben@columbia.edu> and Mark Kennedy <mtk@ms.com>
Vicks and Novell collaborate on VapoWare

3rd and 2nd Places

James Buszard-Welcher <jwelcher@ssf-sys.dhl.com>
Cisco and The Tidy-Bowl Man develop Roto-Router;
Steven Spielberg Productions and Sun Microsystems release Jurassic Sparc

1st Place

Jason Levitt <jason@cs.utexas.edu>
Bell Labs and Lorena Bobbitt announce Plan 4.5

The Judges (in no particular order):

Ed Gould
Matt Blaze
Jeff Haemer
Nick Stoughton
Berry Kercheval
JI
Greg Rose
Piglet

Available Software

by Jeffrey Mogul

<mogul@pa.dec.com>

I asked all the authors of Winter '94 USENIX refereed papers to provide me with information about any software, documents, etc. they are making available via public FTP. (Thanks to Adam Moskowitz for suggesting that I do this.) The list below shows the results.

Please respect any restrictions that the authors have placed on the distribution of their software.

Joshua Auerbach <jsa@watson.ibm.com>

USENIX paper:

Concert/C: A Language for Distributed Programming

Availability: now

Anonymous FTP site name: software.watson.ibm.com

File name(s) and description(s):

/pub/concert/cncrt.B2.tar.Z — (latest Concert /C release also includes essential documentation)

/pub/concert/README — Terms and conditions

/pub/concert/DISCLAIMER — Terms & conditions

/pub/concert/USER_AGREEMENT — Terms & conditions

/pub/concert/doc/index.txt—Index of documentation

/pub/concert/doc/*.txt— Documents described in index.txt

/pub/concert/doc/*.ps —Documents described in index.txt

Fred Douglass <douglass@research.panasonic.com>

USENIX paper: **Thwarting the Power-Hungry Disk**

Availability: now

Anonymous FTP site name: ftp.mitl.research.panasonic.com

File name(s) and description(s):

pub/douglass/thwarting.ps.gz— revised version of paper, incorporating errata from proceedings and different trace.

Steve Summit <scs@eskimo.com>

USENIX paper:

Filesystem Daemons as a Unifying Mechanism for Network Information Access

Availability: 5 February 1994

Anonymous FTP site name: eskimo.com

File name(s) and description(s):

/scs/oofs/ —directory containing Oofs code and daemons

/scs/oofs/README—file describing Oofs distribution; read for further index information

Kurt Lidl <lidl@uunet.uu.net>

USENIX paper:

Drinking from the Firehose: Multicast USENET News

Availability date: 18 Dec 1993

Anonymous FTP site name: ftp.uu.net

File name(s) and description(s):

/networking/news/muse/muse-client.tar.Z

—Client side software to receive news via the Mbone.

Richard Black <Richard.Black@cl.cam.ac.uk>

USENIX paper:

Experience and results from implementation of an ATM socket family

Availability date: future

Anonymous FTP site name: ftp.cl.cam.ac.uk

File name(s) and description(s):

Source code (file names unknown at the moment; contact Richard Black via email for more information, or poke around on ftp.cl.cam.ac.uk)

I took the following information from the papers themselves; these entries were not supplied to me directly by the authors, so don't count on the information in them.

Title: **The refdbms Distributed Bibliographic Database System**

<goldring@cs.vu.nl>

Availability: now

Anonymous FTP site name: ftp.cse.ucsc.edu

File name(s) and description(s):

/pub/refdbms —entire system

finger refdbms@refdbms.cse.ucsc.edu for list of available databases

Title: **ACID: A Debugger Built From A Language**

<philw@research.att.com>

Availability: now

Anonymous FTP site name: research.att.com

File name(s) and description(s): dist/plan9man

— contains manual pages for ACID and ALEF

Board Meeting Summary

by Ellie Young
<ellie@usenix.org>

Below is a summary of the actions taken at the regular quarterly meeting of the USENIX Board of Directors which convened in San Francisco, California on January 16, 1994.

Attendance

Rick Adams, Eric Allman, Tom Christiansen, Lori Grob, Steve Johnson, Evi Nemeth, Mike O'Dell, Barry Shein, Diane DeMartini, Judy DesHarnais, Dan Klein, Ellie Young, Dan Appelman, Matt Bishop, Keith Bostic, Peter Collinson, Mick Farmer, Dan Geer, Judy Grass, Jeff Haemer, Peter Honeyman, Andrew Hume, Doug Lea, Jay Lepreau, Sue LoVerso, Jeff Mogul, Pat Parseghian, Greg Rose, Margo Seltzer, Henry Spencer, Dave Presotto

Winter '94 Conference

Mogul reported that out of 82 submissions, 27 papers were accepted for the technical sessions. The staff reported that attendance was lower than expected for a San Francisco venue, and that the guest room commitment with the hotel was not met. Young and DesHarnais would negotiate with the hotel regarding this.

LISA '93 Conference

Young reported that final attendance was 1062 which was almost twice the attendance of the previous year, and that the tutorials in particular were well-received and attended.

C++ conference

The program chair, Doug Lea, reported that they had accepted 17 out of the 40 submissions received. The program committee had discussed the future of this event, and considering the decline in number of submissions, they recommended that we consider having a workshop on engineering object technology instead. Lea and Jim Waldo will look into this and possibly come back with a proposal in the future.

Mobile Computing Symposium

It was suggested that we continue to look into co-sponsorship issues with a workshop on this topic being sponsored by IEEE TCOS in December, and continue to look for someone to serve as program

chair for a second USENIX symposium on this topic as well, for sometime in 1995.

Member Survey

It was suggested that we get a quotation from the survey firm to increase the number of people sampled and focus on the reasons why members do not renew and/or re-attend conferences

Long Range Conference Models

Johnson presented several models in response to suggestions during the past year that USENIX no longer sponsor two general conferences (and that there were a variety of reasons for considering this – the difficulty in getting enough high quality submissions, the competition with our smaller symposia for these papers, the declining attendance at the general conferences, etc.) Young provided scenarios and observations for future conference models for 1995 and beyond. These were discussed at length. It was agreed that we should cancel one of our general conferences. The SAGE board will be asked to consider the various issues concerning location and dates for future LISA conferences. There was a preference for having the remaining general conference located in the summer.

Young and DesHarnais were directed to examine the existing hotel contracts, consult with SAGE, look at the finances involved, continue discussions with UniForum about co-locating with them, and come back with scenarios for future models.

Announcements via Email

It was decided that as long as our members and attendees have the option of deciding that they do not want to receive announcements electronically, that sending announcements via email was okay. Young would look into the setup and feasibility of this.

Local Groups

SAGE was asked to provide a proposal if they wished to sponsor local groups.

Policies

In response to a board member's request, Johnson presented the following policy which was adopted:

Sexual Harassment Information Sheet

The USENIX Association prohibits harassment of employees by management, co-workers, members, or others associated with the organization. This includes harassment based on race, gender, national origin, religion, color, marital status, medical condition, disability or impairment, age, and sexual orientation.

Sexual harassment is one type of harassment prohibited both by the Association and by law. It includes unwelcome conduct of a sexual nature by employees and others which has the purpose or effect of creating an offensive or hostile work environment. Requesting sexual favors in exchange for employment or other benefits or creating an environment in which harassment is so severe or pervasive as to create abusive work conditions is also prohibited.

Sexual harassment may result from verbal, visual, or physical conduct. Repeated and unwelcome touching, leering, sexual comments, sexual jokes, and exhibition of sexual pictures are examples of the types of conduct which may constitute sexual harassment.

Any person who believes that he or she has been subject to harassment of any kind should contact the Executive Director or President of the Association. All complaints will be thoroughly investigated and, if warranted, disciplinary action will be taken up to and including the termination of employment of any person violating this policy. The law prohibits retaliation against employees for reporting complaints of harassment.

Employees in California may also contact the California Department of Fair Employment and Housing ("DFEH"), which has responsibility for monitoring harassment in the workplace. The DFEH may seek an administrative hearing before the California Fair Employment and Housing Commission ("FEHC") or file a lawsuit in court. Both the FEHC and the courts have the authority to award damages to compensate employees for injuries incurred because of harassment. Information regarding how to contact the DFEH and the FEHC is posted in the workplace.

A policy concerning tenure and job descriptions for the journal's editor and managing editor was adopted. Discussion regarding the appointment of a one-time review committee for the journal was tabled until the next meeting. A subcommittee of Johnson and Allman was formed to look into adopting a policy concerning the relationship between the Board and persons employed

by the Association, and reviewing the bylaws in general. A subcommittee to look into tutorial speakers's honorarium was formed with Shein, Adams, Klein and Young. It was decided that the USENIX attorney would review all contracts that over \$25,000. Young was asked to provide a revised set of guidelines with regard to our investment policy at the next board meeting.

CitySpace

Young presented a request from the CitySpace project to have USENIX become a sponsor and fund \$7,500 for part of the expenses of having an exhibition of innovative educational digital media projects at the San Francisco Exploratorium. It was approved (see p. 42 for information about this project).

Next Meeting

The next meeting will be held in the Washington, DC area, preferably alongside the SANS conference on April 9, 1994.

Space is Available

Does your company have a product or service that would be of interest to USENIX members? If so, a limited number of pages are now available in this newsletter for advertising.

Please contact Diane DeMartini at the USENIX executive office via phone 510/528-8649, FAX 510/548-5738, or email: <office@usenix.org> for ad rates and availability.

USL and UC Berkeley Reach Agreement

UNIX System Laboratories, Inc. and the University of California, Berkeley have announced they have reached an agreement resolving their disputes. The settlement clears the way for the University to release a new, unencumbered version of the Berkeley 4.4 BSD operating system software, to be called 4.4 BSD-Lite.

Ray Noorda, Chairman of Novell, Inc., which recently acquired USL, called the settlement an "excellent example of what can be accomplished by cooperation between the business and academic communities." Mr. Noorda stated that "the settlement permits the University to accomplish its goals but preserves USL's legitimate interest in protecting its intellectual property."

David Hodges, Dean of the College of Engineering at University of California, Berkeley, said that the settlement "once again allows the University to resume its leading role of providing computer software technology transfer to industry. By providing wide distribution of 4.4 BSD-Lite with minimal restrictions on its use, the University will continue to be the focal point for both software research in and commercial development of truly open systems."

The University of California was one of the earliest licensees of UNIX operating system software, originally developed at AT&T's Bell Laboratories. In the 1980s, Berkeley's Computer Systems Research Group issued a series of "Berkeley Software Distributions" containing modifications to the UNIX software. However, because of licensing restrictions, public access to the source code for many of those modifications has been limited to firms holding licenses from USL, which acquired the rights to the UNIX system from AT&T.

In July 1991, the University issued the "Second Networking Release," also known as Net2, which was intended to make available to the public those portions of the Berkeley Software Distributions which were not subject to license restrictions. However, USL brought a lawsuit against the University, claiming that portions of the release contained restricted material. The University denied USL's claims. It also brought a separate action against USL alleging that USL had violated the terms of its Berkeley Software Distri-

bution, also known as BSD, license agreements by failing to give the University credit for certain material in the UNIX release.

Over the past several months, attorneys and computer scientists representing the University and USL have worked together in an effort to reach a compromise on their disputes. The result of these efforts will be a new, unencumbered version of the latest Berkeley Software Distribution called 4.4 BSD-Lite which will retain virtually all of the functionality of the Second Networking Release along with a number of enhancements from the University's latest 4.4 BSD release.

The settlement restricts further use and distribution of certain files in the Second Networking Release and requires that certain files in 4.4 BSD-Lite include a USL copyright notice. In addition to providing several enhancements, the new 4.4 BSD-Lite Release will replace most of the restricted files and incorporates all the agreed-upon modifications and notices. Thus, 4.4 BSD-Lite will not require a license from nor payment of royalties to USL. The University strongly recommends that 4.4 BSD-Lite be substituted for Net2.

Although it has denied the University's claims, USL has also agreed to affix the University's copyright notice to certain files distributed with future releases of the UNIX system and to give credit to the University for material derived from BSD releases which have been included in the UNIX system.

Copies of the source code for 4.4 BSD-Lite may be obtained from the University at nominal cost. Source code copies and further information on 4.4 BSD-Lite and the restrictions on Net2 may be obtained from the Computer Systems Research Group at (510) 642-7780. Information may also be obtained from USL's licensing offices at 1-800-828-UNIX.

[Editor's Note: USENIX plans to publish the manual volumes for 4.4BSD, which may also contain a CD with the whole set, plus source code. An announcement about this will be forthcoming soon.]

(Another Editor's Note: USL and BSDI, in related news, settled their legal dispute.)

Lifetime Achievement Awards

At its Winter '94 Technical Conference in San Francisco, the USENIX Association presented its second annual Lifetime Achievement Awards to Michael Lesk, for inventing UUCP, and to Van Jacobsen, for his work on making TCP "Industrial Strength." The Association also acknowledged Mike Karels and Dave Nowitz as major contributors to these efforts.

The recipients of this award, known as "Keepers of the Flame," receive an original glass sculpture entitled "The Flame," which was designed by Lewis Olson of Noslo Glass Studios, Corning, New York.

The award was first presented in 1993 to The Computer Systems Research Group, honoring 180 individuals for their contribution to the CSRG effort.

The award originated in 1993 and was designed to recognize intellectual achievement and to honor those who have materially changed the world with their contributions to network technology.

Computing Systems Editorship

by Peter H. Salus
Managing Editor, *Computing Systems*
<peter@usenix.org>

At the end of the current volume, Mike O'Dell will be stepping down from the post of Editor-in-Chief of *Computing Systems*. Mike and I have worked together on CS since the Board approved the idea of a technical journal in 1987. It has been a great pleasure for me to do so. Whatever successes CS has had, I attribute to his taste, humor, breadth of knowledge, and ability to withstand my rudeness and pressure. Mike's imprint is on every one of the 28 issues he has presided over. I and the Association will find it difficult to express our gratitude sufficiently.

The post of Editor-in-Chief of *Computing Systems* will be assumed by Dave Presotto, one of the

principals of the Plan 9 project at AT&T Research. Dave has a Ph.D. from UC Berkeley as well as the distinction of having given three papers at the Portland USENIX Conference (June 1985): Presotto, Pike and Presotto, and Ritchie and Presotto. Dave's sense of humor was visible when he roller-bladed into the Mobile Computing Workshop last August (I managed to explain to him that wasn't what mobile computing meant).

Dave, thank you for taking on this role.

Dave is effectively on board, receiving all submissions and correspondence for the past months. In 7.3 and 7.4, both his name and Mike's will appear on the masthead. I welcome him and leave the Association with the bizarre image of me preserving continuity.

Community News

Peg Shafer's new daughter happily announces: "My name is Ruth Elizabeth Zeltzer and I was born at Brigham & Womens Hosptial on March 3rd, 1994 at 04:10 am. I weigh 6 lbs 14 ozs. My mom, (Peg Schafer) and I are doing fine. Dad (David Zeltzer), well, he'll recover soon now that we're home. Please do not expect me to send out much e-mail right away, as I have to learn how to type first. :-)" <rez@a-girl.bbn.com>

Brendan Kehoe recently suffered a car accident. He

has left the hospital in Pennsylvania and now is spending a week or two in the Spaulding Center at Massachusetts General Hospital in Boston. He will move on to living with his family in Augusta, Maine for six weeks, as a Spaulding outpatient. Reports peg his recovery as truly miraculous — much faster than anyone expected. He isn't reading email regularly yet.

Your editor, **Rob Kolstad**, has assumed the presidency of Berkeley Software Design, Inc.

USENIX Member Benefits

As a member of the USENIX Association, you receive the following benefits:

- Free subscription to *login*: — technical features, system administration tips and techniques, international calendar of events, SAGE News, media reviews, Snitch Reports from the USENIX representative and others on various ANSI, IEEE, and ISO standards efforts, and much more.
- Free subscription to *Computing Systems*, the refereed technical quarterly published with The MIT Press.
- Discounts on registration fees for the large, multi-topic Winter and Summer technical conferences, the System Administration conference (LISA), the C++ conference, and the various single-topic symposia addressing topics such as UNIX Applications Development, Security, Operating Systems, High-Speed Networking, and Mobile Computing — as many as ten technical meetings every year.
- Special subscription rates to the periodicals *Open Computing* (the new name for *UNIX World*) and *The LINUX Journal*.
- Discounts on proceedings from USENIX conferences and symposia and other technical publications.
- Discounts on the USENIX Association book series published by The MIT Press. Now available: *The Evolution of C++: Language Design in the Marketplace of Ideas*, edited by Jim Waldo of Sun Microsystems Laboratories.
- Savings (10-20%) on selected publications from McGraw-Hill, The MIT Press, Prentice Hall, John Wiley & Sons, O'Reilly and Associates, and UniForum.
- Right to join SAGE, the System Administrators Guild

Vendor Exhibitions – Should you Exhibit?

by Peter Mui

<display@usenix.org>

Vendor Exhibitions are held at selected USENIX conferences to showcase products we feel are of interest to our membership. We try to have unique exhibitors that you might not be able to see at another show because of their specialized appeal or the size of their company. Some kinds of companies we like to have exhibit are:

- Regional companies geographically local to our conference site
- Smaller companies with interesting or innovative products
- Divisions of corporations i.e., technology-driven divisions of large companies with a specific interesting product to display

Many USENIX members may work in environments that fit these criteria. Should you or your company be exhibiting at USENIX Conferences? We try to make it as easy as possible for a small company, or a division of a large company, to

exhibit, with none of the hassles associated with going to a traditional trade show and at a fraction of the cost. The exhibitor's fee is scaled to the number of expected attendees, and includes all of the furniture and drapery for tabletop presentation, which is set up in advance.

In 1994, we will be having vendor exhibitions at the following conferences:

Summer Conference, June 6–10
Boston, MA (1500 attendees)

LISA Conference, September 19–23
San Diego, CA (1000 attendees)

Winter Conference, January 16–20, 1995
New Orleans, LA (1000 attendees)

For more information on exhibiting at these shows, contact Peter Mui at the USENIX office: 510/528-8649 (phone), 510/548-5738 (FAX), or send email to <display@usenix.org>.

Living with CC:mail

by Hal Pomeranz

<pomeranz@sclara.qms.com>

[SAGE Editor's Note: Hal Pomeranz asked me about trying a column in which people from throughout the system administration community shared their latest hacks and tricks, those too small to become paper prospects for LISA or SANS, but very useful none the less. From that idea grew this first article. I am still looking for a title for this column, and some more contributors.]

For better or worse, Corporate IS at QMS headquarters decided that CC:mail was to be our corporate email standard. I happen to support the anarchists at the Santa Clara R&D facility, who are happily entrenched in our world of twenty eleven different mail and news readers, but at least they are all based on SMTP/NNTP. Things went along fine for a while — they would send us uuencoded MS Word documents as attachments, and we would send back uuencoded Frame-Maker documents. Nobody was getting any work done, but we were having fun with the whole thing.

Then somebody back at headquarters got the bright idea that since CC:mail was our corporate email standard, they could post important information on the CC:mail bulletin boards and everybody would see it. Now things started getting personal. After mediating enough disputes of the form

"I already said that!"

"Where?"

"On the CC:mail bulletin board."

I decided to be a good guy and save the folks at HQ from their own mistake. Mind you, I waited just long enough that I would be a hero for solving the problem — I am not that big a fool.

So, I talked to Doug, a good guy and the CC:mail wrangler for the R&D group back at HQ. He has already gatewayed the CC:mail bulletin board postings to a mail alias on his UNIX mail server, where he was just dropping it in a file. In case you are wondering, there is no news spool back at HQ: the few folks back there who read news do it from my news server in California.

He also set up an alias that people could send mail to in order to post things on the bulletin boards. He did mention that his alias for postings had to enumerate every CC:mail post office in the

company, because postings from a non-CC:mail host would not get propagated to the bulletin board mirrors at the other post offices. A bug, but one with an obvious workaround.

So Doug can forward me mail messages containing all postings made to the CC:mail bulletin boards, and I can send replies back to his alias. All I need is to gateway mail into news and vice versa, and I know that part has already been done by someone. So, I send mail to my friend Eiji back East whom I turn to for all my news problems. He's starting a new business and isn't responding to email quickly. So, I post to *new.admin.technical* and the moderator sends me pack email in five minutes saying, "'newsgate', written by Rich Salz. Check archie." I love the Internet.

The newsgate software is short on documentation, but was clearly written by someone who understood the problem completely. Newsgate is actually two programs, *mail2news* and *news2mail*, each of which does half of a bi-directional USENET/email gateway. If you have some experience with news software and mailing list maintenance, it's pretty easy to get things going. In case you don't, here's the blow-by-blow description.

First you need to create the newsgroup corresponding to your CC:mail bulletin board. We have an "Announcements" board, so I created a newsgroup *ccmail.ann*. Next you need to create a bunch of aliases. First you need an alias that errors (like mail bounces) should go to. This should resolve to something like "postmaster" or "sysadmin", or whatever alias your email guru hides behind. Next you want to create an alias that posts email to the *ccmail.ann* newsgroup. This should just be a pipeline to *mail2news*.

A small snag came up at this point. I wanted to allow my users to send mail to this alias in order to post articles, so things going to this alias had to propagate back to the CC:mail boards at HQ. If postings were forwarded from the bulletin boards to this alias, however, those postings would also get propagated back to HQ. News handles this problem by having a unique message ID associated with each article (any articles that show up with duplicate message IDs just get junked), but CC:mail has no such feature. I need to put a tag on postings that originate from the bulletin boards so that they don't get back-propagated and posted twice.

No problem, *mail2news* allows me to set the "Organization:" header. I create two aliases: "ccmail.ann" for my users and "ccmail.ann.bb" for automatic mailings generated by postings to the bulletin boards. Each alias sets the Organization field to something different and I can filter based on that. The whole aliasing scheme ends up looking like

```
control: postmaster
ccmail.ann: "| /usr/lib/news/.admin
/mail2news
-o 'q.m.s.mail/.news.gateway'
-n.ccmil.ann"
ccmail.ann.bb: "| /usr/lib/news
/.admin/mail2news
-o 'c.c.mail.gateway' -n ccmil.ann"
```

The -o flag is used to specify the Organization: field – a dot before a letter means capitalize that letter in the final posting (i.e., "QMS Mail/News Gateway" and "CC:mail Gateway")– and -n specifies the newsgroup. Simple.

At the other end, I now need to mail postings to the newsgroup back to the alias at HQ which explodes the posting out to all the CC:mail post offices. This is tricky and depends heavily on what news software you are running. Here's the entry from my /usr/lib/news/sys file so you can get the basic idea

```
gateway:ccmail.ann/all:
/usr/lib/news/.admin/filter
/usr/lib/news/.admin/news2mail --
ccmail.ann ccmil-ann control qms.com
```

"filter" is a small Perl script I wrote to discard any articles which match "CC:mail Gateway" in the Organization: field. It pipes all other articles to the *news2mail* command line given on its argument list. If you don't want to write "filter" yourself, see the end of this article for more information.

news2mail plays tricky games with mail headers based upon the arguments it receives. The second argument (ccmail.ann) ends up in the Reply-to: field of the mail. I actually have a mailing list of users who receive postings via mail, so I want their replies to get posted back to the newsgroup. The third argument is the alias which explodes the posting to all of the CC:mail bulletin boards in the rest of the company – it's the To: field. The fourth argument gets dropped into the Sender: field, so any errors end up going to the "control" alias. The final argument is my domain name. See, I told you that Rich knew what he was doing.

Okay, what's the "--" about? *news2mail* was designed to handle high volume newsgroups. It puts it's mail messages into the sendmail queue rather than trying to send them immediately. For my application, the volume was quite low, and users wanted instant gratification, so I hacked. The "--" says "send the mail directly rather than queueing." For patch availability information, see below.

All done and all I did was install two new programs, write a 10-line Perl script, make a 5-line modification to some source code, create a few aliases, and modify one news configuration file. Well, okay, maybe I did earn my money that day, but it will be easy for you.

Now you're excited and want to try this out immediately. The text of this article, a set of terse instructions, my filter script, and my patch to *news2mail*, along with a copy of the *news2mail* source code are all available from ftp.qms.com (161.33.3.1) in the directory /pub/ccmail. Also in there, find a little Perl filter called "undamage" which unpacks uuencoded CC:mail attachments (like MS Word documents) and extracts and reformats the text. Invaluable if you have CC:mail in your company but you don't have a PC on your desk.

Return to the Octopus...

by Elizabeth Zwicky

<zwicky@erg.sri.com>

Since the publication of my column on the squeaky octopus, a number of people have written to me asking where to get a squeaky octopus. My original squeaky octopus came from a party store, where it was being sold as an indispensable adjunct for your Caribbean theme party, along with paper pineapples. More logically, they are widely available on piers in California, along with stuffed otters and sea lion food. If your locality boasts neither piers nor peculiar party stores, there are several options.

First, you can use other squeaky objects; we have tested squeaky lizards, frogs, penguins, and pickles. I refused to have anything to do with the squeaky killer whale, which seemed an inherently confused concept to me, and the squeaky tarantula failed the test for similar reasons, although the fact that it spent the time before Halloween being sprung on people probably also contributed to its failure. At any rate, none of our users will touch it. Any toy store or party store worthy of mention will surely provide you with a range of squeaky choices, and our testing appears to show that you should be fine as long as you steer clear of venomous animals.

If you are attached to the squeaky octopus concept, there is at least one mail-order squeaky octopus vendor, Archie McPhee. Their address is P.O. Box 30852, Seattle WA, 98103, USA, and their phone number is 206-782-2344. They also supply a number of other mind-boggling objects which may have hitherto unsuspected system administration uses, like Chinese punching puppets and glow-in-the-dark slugs.

System Administration Tools Your Vendor Never Told You About: The Telephone Headset

by Elizabeth Zwicky

<zwicky@sri.erg.com>

Most system administrators spend a staggering amount of time on the telephone. Much of this is time spent trying to debug a problem along with the person on the other end of the phone, or time spent trying to talk to a user while also using a keyboard for something (playing tetris or reading news, for instance).

In this situation, a telephone headset is a major asset. Suddenly, you're not spending much of your day viewing the screen at an angle because you're using your head to pin the receiver against your shoulder so that you can type. Phone stores and really big office supply stores carry them. It helps if you can play with them before you buy, because not everybody's head is the same size, and you want to be able to put the mike conveniently close to your mouth.

If you occasionally have long conversations, but are usually happy with a handset, look for one that's easy to put on and turn on. If you answer the phone a lot, look for a good sturdy quick-release clip; you're just going to put it on and leave it there, but you want to be able to plug and unplug it while you go for a cup of coffee. (Yes, you look funny wearing a headset with a dangling cord, but you look even funnier with a permanent twist in your neck.) *[Ed. Note: I gave up my handset three years ago.]*

Perl Practicum

"I'm Beginning to See a Pattern Here"

by Hal Pomeranz

<pomeranz@sclara.qms.com>

The last two Perl Practicum articles may have strayed a bit from the true path, so let us return to some Perl basics. Regular expressions are a Perl fundamental, but many people seem to have trouble thinking in regular expression mode. This issue will give you some basic strategies for not becoming overwhelmed in a soup of funny looking characters.

The first basic rule is, always take full advantage of naturally occurring delimiters. We put spaces between words in written (and spoken) English because it helps us to understand it better -- look for fixed tokens that help you break up your regular expression "utterances". For example,

```
/^[-+]?[0-9]+(\.[0-9]+)?([eE][-+]?[0-9]+)?$/
```

is just so much Greek if you try to read it all at once. Use the () and [] groupings to break the expression up into four manageable pieces:

```
[+-]? [0-9]+ (\.[0-9]+)? ([eE][-+]?[0-9]+)?
```

The first one is easy, an optional plus or minus sign, and the second is trivial, one or more digits. The third says, "a literal period followed by one or more digits", and the trailing question mark makes the whole group optional. The fourth (also optional) group is a little trickier: an upper or lower case 'E', followed by an optional plus or minus, followed by one or more digits. Put it all together and you match any valid Perl number, but you probably figured this out by now.

This rule should also be applied when building up regular expressions. Suppose we wanted to match date strings

```
Fri Jan 28 13:12:02 PST 1994
```

There are six different space separated blobs in that line, but there are only two fundamental "types" of things to match: words ("Fri", "Jan", and "PST") and numbers ("28", "1994", and the hours, minutes, and seconds in the time string). Well we can just use "\w+" for words and "\d+" for numbers, and the regular expression just pops out

the expression below is wrong!

```
/^ \w+ \w+ \d+ \d+:\d+:\d+ \w+ \d+$/
```

Actually, this is not quite right. The day of the month and the hour of the day can both be single digit values, and the leading digit position will then just be a space. So, we modify our pattern slightly

```
/^ \w+ \w+ \s+ \d+ \s+ \d+:\d+:\d+ \w+ \d+$/
```

I generally find "\s+" clearer than " +" (that's space-plus, see what I mean?) in regular expressions, even though they don't strictly mean the same thing.

The process we used to build up the last example brings us to our second rule: start simple and increase your complexity and level of refinement gradually. For example, it was my recent misfortune to have to parse a file with lines like

```
Pomeranz, Hal (pomeranz) x409
```

Sometimes the white space was literal spaces, sometimes tabs, other times a mixture of the two, and there tended to be lots of trailing white space. Sometimes there was no email address, sometimes there was no extension, and sometimes there was neither.

A first cut might be

```
/^ \w+, \w+ \s+ (\w+ \s+) x\d+$/
```

You can clearly see the four blocks corresponding to last name, first name, email, and phone extension. Note that we have to backwhack the parentheses around the email address because of their special meaning in regular expressions. Now we can begin to address special cases.

The email address and phone extension are optional

```
/^ \w+, \w+ ( \s+ (\w+ \s+) )? ( x\d+ )?$/
```

Note that we have incorporated the space before the email address and phone extension in the optional block along with each of those fields. Theoretically, the line of data could simply end after the first name with no additional white space. As a further refinement, we have to deal with trailing white space, and the case where field delimiters are not single spaces

```
/^ \w+, \s+ \w+ ( \s+ (\w+ \s+) )? ( \s+ x\d+ )? \s*$/
```

Actually, last names can look like "Van Der Sluis" or "Cody-Lang", so we remember Rule #1 (take advantage of naturally occurring delimiters) and say that the last name is anything before the comma

```
/^ .+, \s+ \w+ ( \s+ (\w+ \s+) )? ( \s+ x\d+ )? \s*$/
```

All right, we know the above expression accurately matches all the data we might encounter

because we have tested it thoroughly on actual data (you did test thoroughly, right?). Actually, I really needed this pattern so that I could extract the last and first names, email address, and phone extension from the line. So now we have to make everything we want to extract from the line into a subexpression by throwing parentheses around the individual fields

```
/^(.+),\s+(\w+)(\s+(\w+)\s+)?(\s+x(\d+))?\s*$/
```

As Randal Schwartz is fond of saying, "Perl: checksummed line noise with a sense of purpose."

The third rule is, never use a complex expression when a simple one will do. For example, one expression to match IP addresses might be

```
/^([12]?[0-9]?[0-9])\.([12]?[0-9]?[0-9])\.
```

but why bother? In most cases either

```
/^\d+\.\d+\.\d+\.\d+$/
```

or

```
/^(\d+\.){3}\d+$/
```

is more than sufficient. The first expression is probably more readable, but your mileage may vary. In either case, the person who has to maintain your code six months from now (who, you should remember, might just be yourself) will thank you.

Rule number four is never forget that Perl pattern matching is greedy: the '*' and '+' operators will eat as much as they can as long as the pattern can be satisfied. This can work in your favor when you are doing something like

```
$_ = "/usr/local/bin/perl";  
($dir, $prog) = ~/^(.*)\/(.*)$/;
```

The first ".*" will eat up everything but the last '/' which we force it to match (Rule #1 again) before we pull off the program name.

This greedy behavior can be a problem as well, particularly when you are trying to match pairs of delimiters. For example, suppose you wanted to match the first double quoted field in

```
$_ = 'pomeranz "Hal Pomeranz" "S Clara"';
```

The expression

```
$name = ~/\"(.*)\"/ # wrong!
```

will set \$name equal to

```
Hal Pomeranz" "S Clara
```

which is not what we wanted. Instead you want

```
$name = ~/\"([^\"]+)\\"/
```

which says match a double quote, followed by one or more things that are NOT a double quote, terminated with another double quote. This "match everything except my trailing delimiter" concept is a useful trick for your Perl toolkit.

The fifth and final rule is, be careful about anchoring your patterns with ^ and \$. Err towards using ^ and \$, even when they are not strictly necessary. For example, a common idiom is

```
@files = grep(!/^\.\/\.$/, readdir(DIR));
```

which gives you a list of files from directory handle DIR, except for the "." (dot) and ".." (dot-dot) files. Leaving off the ^ and \$ accidentally will throw away all filenames with a dot in them, and leaving off the \$ will throw out all dot files in the directory. Either way, the result is bound to be unexpected.

Another place where this can bite you is when you are trying to verify the format of some data. The pattern

```
^\d+/
```

will match valid integers, but it also matches "foo2bar" and other things which are definitely not numbers. To validate that values are numbers you have to use

```
/^\d+$/
```

or a more complex expression like the one at the beginning of this article.

You simply must become comfortable with regular expressions to use Perl effectively. Always remember to break complex regular expressions up into manageable pieces before trying to write or understand them. Always work up from a simple case to greater stages of refinement and complexity. Never make expressions any more complex than they have to be or you will never be able to modify them without breaking something else. Use greedy pattern matching to your advantage but beware of the dark side. Finally, use ^ and \$ freely to avoid unexpected problems.

PLATO: The Emergence of On-Line Community

by David R. Woolley
<drwool@well.sf.ca.us>

[Editor's Note: This article was first published in the Matrix News which has provided permission to reprint it here.]

The PLATO system was designed for Computer-Based Education. But for many people, PLATO's most enduring legacy is the on-line community spawned by its communication features.

PLATO originated in the early 1960's at the Urbana campus of the University of Illinois. Professor Don Bitzer became interested in using computers for teaching, and with some colleagues founded the Computer-based Education Research Laboratory (CERL). Bitzer, an electrical engineer, collaborated with a few other engineers to design the PLATO hardware. To write the software, he collected a staff of creative eccentrics ranging from university professors to high school students, few of whom had any computer background. Together they built a system that was at least a decade ahead of its time in many ways.

PLATO is a time-sharing system. (It was, in fact, one of the first time-sharing systems to be operated in public.) Both courseware authors and their students use the same high-resolution graphics display terminals, which are connected to a central mainframe. A special-purpose programming language called TUTOR is used to write educational software.

Throughout the 1960's, PLATO remained a small system, supporting only a single classroom of terminals. About 1972, PLATO began a transition to a new generation of mainframes that would eventually support up to one thousand users simultaneously.

PLATO Notes: Original Development

In the summer of 1973, Paul Tenczar asked me to write a program that would let PLATO users report system bugs on-line. Tenczar was the head of the system software staff, and I was a 17-year old university student and junior system programmer. I had been with CERL for about a year, learning the ropes and doing minor programming tasks at minimum wage.

We already had a way for users to report bugs, but it was just an open text file called "notes".

Anyone could edit the file and add a comment to the end. After investigating a problem, a system programmer would insert a response (typically something like "+++Fixed - RWB").

This was simple enough, but there were problems. For one thing, only one person could edit the file at a time. For another, there was no security at all. It was impossible to know for sure who had written a note. Most people signed or at least initialed their comments, but there was nothing to enforce this. And occasionally some joker would think it was fun to delete the entire file.

It was just such an incident that prompted Tenczar to ask me to develop a replacement. His idea was a simple refinement of the method we had been using: a user would type a problem report into a special-purpose program, which would automatically tag it with the date and the user's ID and store it safely in a tamper-proof file. The same program would allow convenient viewing of the stored notes. Each would appear on a split screen, with the user's note on the top half and the system staff's response below.

It occurred to me that half a screen might not be enough space for some notes. And that some problems might require back-and-forth conversation between a user and the system staff. A limit of one response per note wouldn't permit much dialog.

I came up with a design that allowed up to 63 responses per note, and displayed each response by itself on a separate screen. Responses were chained together in sequence after a note, so that each note could become the starting point of an ongoing conversation. This is what John Quarterman calls a star-structured conferencing system, and PLATO Notes was apparently the first of its kind.

My first prototype kept all notes in one file. Upon entry you would see an index of the most recent notes, listing each note's number, date, title, and number of responses. You could then select a note to read, or page back through the index to find older notes.

As I showed this to other members of the system staff, we began to talk about other ways that this program might be used beyond just problem reports. We thought it would be nice to have a separate area where new users could ask ques-

tions and get help from more experienced users, and another area where the system staff could announce new PLATO features. So I added a top-level menu to let people choose among three notesfiles: System Announcements, Help Notes, and General Notes.

Notes was released on August 7, 1973. It was named after the text file it replaced, so that people accustomed to typing "notes" would be taken to the right place.

Every note or response appeared on its own screen. Since PLATO was designed for education, its architecture was biased toward carefully crafted full-screen displays. It was easy to place text or graphics at specific locations on the screen, but nearly impossible to scroll text. For Notes, this was both an advantage and a drawback. One nice feature was that the note title, date, time, and author's name always appeared in the same place. After using Notes for a while, your eye "knew" exactly where to look for these things.

On the down side, each posting was limited to 20 lines of text so as to fit on one screen. The only way to overcome this was to write a series of responses, but that allowed other responders to slip in and disrupt the flow. Still, the 20-line limit had the virtue of encouraging brevity.

Most options for reading notes required only a single keypress. While reading a response, for example, one keypress could perform any of these functions (among others):

- proceed to the next response
- go back to the previous response
- go back to the base note
- skip to the next base note
- begin writing a new response

There were too many options to list them all on every screen. Most prompts were quite minimal, but a Help key was universally available. It would display a complete list of the options available at any point.

Notes quickly became an indispensable part of the landscape. It appeared just as PLATO was beginning a phenomenal growth spurt made possible by the new mainframe. Although PLATO had been evolving for over a decade by this time, to the new flood of users coming on-line, PLATO without Notes was hard to imagine.

The PLATO Architecture

PLATO is designed to be extremely responsive to keys. Every keypress is processed individually by the central mainframe, but the response (or

"echo") is usually so fast as to appear instantaneous. An echo time of 100 milliseconds is excellent; anything over 250 is considered unacceptable.

This is vital, especially because displays do not appear instantaneously. Originally, all PLATO terminals communicated at 1200 bps. At that speed, a long posting in Notes might take up to 10 seconds to fill the screen. But a single keypress aborts the display and moves on if the first line or two of a note doesn't spark your interest.

The ability to abort pending display output is crucial. Even now that faster connections are possible, connecting through a network that does not permit aborting output makes PLATO feel maddeningly sluggish.

Talkomatic and "Term-Talk"

Any competent PLATO programmer can quickly hack together a simple chat program that lets two users exchange typed one-line messages. PLATO's architecture makes this trivial. A few such programs existed on PLATO before 1973, but they did not get much use, probably because the user community was quite small and most terminals were still in a single building.

In the fall of 1973, Doug Brown designed a program that let several users chat as a group. He wrote a simple prototype to demonstrate the concept and called it Talkomatic.

The real magic of Talkomatic was that it transmitted characters instantly as they were typed, instead of waiting for a complete line of text. The screen was divided into several horizontal windows, with one participant in each. This let all the participants type at once without their messages becoming a confusing jumble. Seeing messages appear literally as they were typed made the conversation feel much more alive than in line-by-line chat programs.

I worked with Doug to expand Talkomatic to support multiple channels and add other features. Each channel supported up to five active participants and any number of monitors, who could watch but couldn't type anything. (One drawback to the Talkomatic approach is that the size of the screen limits the number of participants in a channel.)

Empty channels were open to anyone, but any active participant in a channel could choose to "protect" it. This prevented anyone from monitoring the channel, and the participants could then decide who else to admit.

Talkomatic was an instant hit. Soon it was logging over 40 hours of use per day. It was not officially part of the PLATO system software, and in fact it was used mostly for what administrators would consider frivolous purposes. There was no way to contact a specific person to let them know you wanted to talk, so it was more like a virtual water cooler than a telephone substitute. People would hang out in a channel and chat or flirt with whoever dropped by.

But Talkomatic was so appealing that it inspired the system staff to create an officially supported chat feature. It became known as "term-talk" because it could be accessed from anywhere on PLATO by pressing the TERM key and typing "talk". The TERM key was originally meant to provide hypertext-like branching to term definitions. In practice, it was rarely used for terms, but it was handy for instant access to features like "talk".

A "term-talk" conversation was limited to two people, but had its own advantages: you could page a specific person, and you could use it without exiting from whatever else you are doing. A person receiving a page would see a flashing message at the bottom of the screen identifying the pager, and could use "term-talk" to accept. The bottom two lines of the screen then became a miniature Talkomatic. An unwanted page could be rejected with "term-busy", or simply ignored until the pager gave up.

A feature was later added to "term-talk" that allowed the participants to switch to "monitor" mode, in which one person could actually view the other's screen. The person being monitored was free to move about the system normally, editing files, running programs, etc. This was extremely useful for remote consulting: someone who needed help could literally show an on-line consultant what they were trying to do while maintaining a conversation at the bottom of the screen. To ensure privacy, monitor mode could be initiated only by the person whose screen was to be monitored.

Personal Notes

With Notes and "term-talk" in place, it began to seem natural to use PLATO as a means of communication. What it obviously lacked was a way to send private mail. Kim Mast tackled the job, and in August of 1974, Personal Notes was released.

Personal Notes was similar to Notes in many ways: each note appeared on a separate screen, and options such as moving to the next or previous note, deleting a note, or responding, were

available as single key presses. There was no index of notes, however. Entering Personal Notes took you immediately to the first note you had not yet read. From there, you could move forward or backward through your notes.

Kim and I worked together to integrate all of PLATO's communication features into a seamless package. For example, while viewing a note, you could:

- copy it to a notesfile
- forward it to someone as a personal note
- send a personal note to the author
- initiate a "term-talk" with the author

All of these options were available in both Notes and Personal Notes, and the same keys were used in both.

Notes Categories

The success of Notes led to overcrowding. There were only two notesfiles which users could write in, and they were used for practically everything. It became a chore to wade through the volume of notes written every day, and people began to ask for a way to filter out notes they weren't interested in.

My solution was this: the system staff would define a list of categories, such as "bug reports", "suggestions", "events", "jokes", etc. Anyone writing a note would assign it to one of these categories. Users could select which categories they wanted to see when reading notes, and their selections would be stored permanently as part of their user records.

In early 1975 I created a version of Notes that supported categories, and released it to a limited group of users for testing and comments.

Suggestions from users were vital to PLATO's evolution, and Notes was no exception. Since I had written Notes originally, it was my turf, and I made most of the decisions about what features were implemented. But I had the benefit of lots of ideas from users as well as from the rest of the system staff. Often a suggestion would strike me immediately as great idea, and if it was not too difficult, it might be implemented and released within a day or two. Not all ideas were implemented, by any means. But sometimes I would argue against a proposed change, only to be convinced of its merit by cogent arguments or by the sheer number of people voicing support for it. Thus, Notes was shaped largely by a consensus of the entire PLATO community.

The notes categories concept was well received at first, but it got bogged down in controversy over features and never made it to general release. A particularly contentious issue was how notes should be presented. One faction wanted to see all notes in chronological order, with the categories serving only as a filter to skip unwanted notes. Others wanted categories to serve an organizing function, as well: all the notes from one category would be shown, then all the notes from the next category, and so on.

Strange as it seems now, I held out against organizing notes by category. I was used to reading notes about many different subjects all jumbled together, and just wanted to be able to see all the new notes listed together in one place. But support for more organization grew, and I began to see that I was in a losing battle.

In the meantime, though, other problems became apparent. First, I realized that as the volume of notes increased, there would be technical problems with keeping everything in one large file. Second, it wasn't clear how many categories would be needed. I had designed in a limit of 60, which seemed like a vast increase over the 3 we had been living with. But if we ever needed more it would be very difficult to increase the limit. After months of wrangling, my concept of notes categories seemed fatally flawed. I really didn't know where to go with it.

About this time, a few people began to ask for private notesfiles. We had all seen how useful Notes was for discussing development of PLATO itself. Couldn't the concept be extended to allow any small group of people working on a project to communicate among themselves? In fact, a group in Chicago that was using PLATO to develop medical courseware wrote a clone of Notes for their own use.

Suddenly the future clicked into focus. I abandoned the categories project and began to implement Group Notes.

Group Notes

Group Notes was a generalization of the original Notes. Now there could be an unlimited number of notesfiles, and users would be able to create private notesfiles for use by their own work groups. Group notesfiles would serve the same purposes for which notes categories were designed with none of the inherent problems. The 60-category limit vanished. Distributing notes across many files solved the technical problems of dealing with large volumes of information. The burden of managing notesfiles would be distrib-

uted, as well; no longer would the system staff have to oversee everything. And, yes, notes would be organized by subject, as so many people had insisted. Group Notes is one of those ideas that, with hindsight, seems glaringly obvious.

Group Notes was released in January, 1976, and thereafter use of Notes skyrocketed. Soon there were public notesfiles for subjects like books, movies, religion, music, and science fiction, as well as many private notesfiles for work groups.

The internal structure of notesfiles still had not changed much since 1973, and it was beginning to show its weaknesses. In particular, it made it difficult to implement a sorely needed option to read all responses written since a certain date and time. So I rewrote Notes almost from scratch, and converted all notesfiles to a new internal structure in July, 1976. Although it has been modified many times since, this version forms the core of the Notes software still in use today.

Access Lists

Access lists are the key to Group Notes. A person who creates a notesfile is automatically registered as a "director" of the file. A director can edit the access list to specify who else can access the notesfile and with what privileges. Access can be specified for individual user IDs or for entire work groups, and any level of access can be granted to the general public (anyone not specifically listed).

There are six access levels:

- Director
- Read/write
- Read/respond
- Read-only
- Write-only
- No access

Read/write is the most common type of access. It permits both writing new notes and responding to existing notes.

Read/respond permits responding, but not writing new notes.

Write-only access permits a user to write new notes, but not read or respond. It is sometimes used as a blanket access level for the public, providing a way for someone to request access to a private notesfile. It is also useful for collecting comments from the public about some issue, while maintaining the privacy of each person's remarks.

Generally, anyone who can read a notesfile can also view its access list, although the director can choose to prohibit this.

Reading By Date

Notes offers a way to read all notes and responses written since a certain date and time. This feature is designed so that you can sequence through all new postings using a single key. For every note with new responses, the base note is displayed first to provide context. A keypress then skips to the first new response. Pressing the same key repeatedly sequences through the rest of the response chain, and then skips to the next note with new responses.

In 1978, John Matheny implemented the Notes Sequencer, a great boon to habitual notes readers. The Sequencer lets you create a personal list of the notesfiles you read regularly, and automatically keeps track of the last time you read each one. Using the Sequencer, you can quickly scan all the notesfiles in your list for new postings with a minimum of key presses.

Deleting Notes

Someone who has second thoughts after posting a note or response can delete it or edit it, as long as no responses have been added after it. This restriction is meant to avoid garbling the thread of a conversation. Deleting a response from the middle of a chain can make the following responses seem nonsensical. But an author who desperately wants to delete a posting anyway can appeal to the notesfile director, who can delete any posting without restriction.

A director can delete a response from the middle of a chain without disturbing subsequent responses. However, if a director deletes a base note, all responses disappear with it. Directors frequently use this power to clean out a notesfile, removing old notes that are no longer of interest.

Anonymity

The idea of anonymity in Notes was controversial when first proposed, but the issue was resolved by leaving it to the discretion of each notesfile director. If a director chooses to allow anonymity, then anyone posting a note or response in that notesfile is given the option of making it anonymous.

An anonymous note is truly anonymous. Not even the notesfile director nor the system staff can determine who posted it, because the user ID is not saved anywhere. The word "anonymous"

appears in the header where the user ID would normally be.

PLATO Notes avoids some of the problems that have plagued experiments with anonymity in other conferencing systems. It is not possible to masquerade as someone else, because Notes does not allow the use of pseudonyms. The only identification that can appear in the header is the author's actual user ID or the word "anonymous". The fact that anonymity is the choice of each user is important, too. Someone could post an anonymous note saying "I'm David Woolley and I kick my dog every morning," but everyone reading it knows that the author specifically chose to make this note anonymous, so the identity claimed in the text is not to be taken seriously.

Most notesfile directors do not permit anonymity, but it is very useful in some situations. Anonymity can be abused, but a notesfile director can delete offensive postings. The version of Notes now used on NovaNET even allows a director to review anonymous postings before they become publicly visible.

Director Messages

Another privilege that notesfile directors have is to flag their postings with a "director message," a single line of text which appears above the standard header. Directors often use the message to flag official postings, such as statements about policy or notices that an inappropriate note was deleted. The director can specify what the message should say, but a single message has to suffice for all situations since there is only one director message per notesfile. Typical messages range from the serious ("OFFICIAL MESSAGE") to the humorous ("Not Operating With A Full Deck").

A director can toggle the message on or off for any posting, even those written by other people. For example, in a notesfile used to report problems, a director might set the message to "FIXED" and use it to flag problems that have been resolved.

"Term-Comments"

One of the ways that Notes supports PLATO's educational purposes is through a feature called "term-comments". While running a program, a user can press TERM and type "comments", and then type a note to the program author. Such comments are collected in a notesfile that the author has associated with the program. Each note is tagged with a header indicating the exact point in the program where the comment was

made, so if a note reports that "entropy is misspelled on this page," the author knows exactly where to look for the error.

Linked Notes

Around 1975, Control Data Corporation set up its own PLATO system in Minneapolis and began turning PLATO into a product. By 1985, over 100 PLATO systems were operating at sites around the world, about 60% of them running full-time. Some of them were linked together with dedicated lines so that files and notes could be exchanged easily. Both Group Notes and Personal Notes were modified to support intersystem links in 1978.

A notesfile can be linked between any number of systems. From a user's viewpoint, a linked notesfile is exactly like any other, except that the notice "Linked Notesfile" appears on the index display, and in the headers of some postings a system identifier appears after the author's user ID.

When a note or response is posted in a linked notesfile, it appears immediately in the local copy of the file, and is put in a queue to be broadcast to all systems which share that notesfile. The Notes software does its best to keep the file identical on all systems, although it can't guarantee that responses in a given chain appear in exactly the same order everywhere. There can be a delay of several minutes to an hour before a response is posted on linked systems (or even longer if one of the systems is down for an extended period).

Star Structured vs. Tree Structured Conferencing

Although Notes has evolved in many ways over the years, one thing that has never changed is the star structure of its notesfiles. One or two PLATO users wrote experimental versions of Notes using tree-structured notesfiles, but most people who tried them found them hard to use and the idea did not catch on.

My own feeling is that a star structure is much more conducive to ongoing discussion. Human conversation is inherently disorganized, and a tree structure attempts to impose too much discipline. Conversations often tend to fragment and dissipate quickly in a tree. Some people seem at home with a tree structure, but in my experience more people find it rather foreign and overly complex.

With a star structure, each base note and its chain of responses resembles a conversation that we might have with a group of people gathered around a table. The conversation might drift or develop multiple threads, but if that becomes a

problem, it is easily dealt with by simply starting new base notes to carry on divergent threads. A notesfile director can suggest this, but often the participants do it themselves.

Multiplayer Games

There are myriad games on PLATO. Some are for single players, but the most popular ones involve two or more players at separate terminals.

Games were certainly not a priority when PLATO was designed, but it turns out that its architecture supports multiplayer games superbly. The crucial features are:

- shared memory areas
- standardized terminal
- high resolution graphics display
- central computer processing of every key
- fast key response
- ability to abort display output

Rick Blomme wrote PLATO's first two-player game in the late 1960's, a simple version of MIT's Spacewar. Possibly the most popular game in PLATO history is Avatar, one of several dungeons n' dragons games. Empire, a multiplayer game based on Star Trek, is another favorite. Other multiplayer games range from Airfight (a precursor to Microsoft Flight Simulator), to Wordwar (a spelling and speed-typing game) and card games such as contract bridge.

Most games were written by unpaid programmers. The only reward they could hope for was the prestige of having written a popular game. Some game authors now receive royalties, but it amounts to only a few cents per hour of use, often split between a number of co-authors. A number of games that originated on PLATO have been recreated commercially as video arcade or personal computer games.

The On-Line Community

The sense of an on-line community began to emerge on PLATO in 1973-74, as Notes, Talkomatic, "term-talk", and Personal Notes were introduced in quick succession. People met and got acquainted in Talkomatic, and carried on romances via "term-talk" and Personal Notes. The release of Group Notes in 1976 gave the community fertile new ground for growth, but by that time it was already well established. The community had been building its own additions to the software infrastructure in the form of multiplayer games and alternative on-line communications. One such program was Pad, an on-line bulletin board where people could post graffiti or random musings. Another was Newsreport, a lighthearted

on-line newspaper published periodically by Bruce Parelo, aka The Red Sweater.

With the abundance of special interest notesfiles made possible by Group Notes, many on-line personalities developed. One of the best known was Dr. Graper (actually a student at the University of Delaware named David J. Graper). He began posting wild, surrealistic stories in a public notesfile where they were not exactly appropriate, but they were so hilariously entertaining that people clamored for more, and eventually someone created a notesfile called Grapenotes as a platform for his ravings.

The early PLATO community was concentrated in Illinois and consisted mostly of people in academia: educators turned instructional designers, and students hired as programmers. Later it grew to include more people from business, government, and the military as Control Data marketed PLATO as a general-purpose tool for training. It also grew geographically, spreading across the United States and around the world. The building that housed CERL became something of a Mecca to the far-flung PLATO community. Many people traveled to Urbana to see the lab and meet those of us who worked there. It was odd to meet people face to face after getting to know them on-line. My images of people based on their postings in Notes sometimes turned out to diverge wildly from reality.

The growing PLATO community also developed all of the problems that are now well known in on-line communities, such as flaming, men impersonating women as a prank, etc. Free speech was the general rule, but there were a few much-discussed incidents in which political postings in notesfiles were officially quashed for fear of jeopardizing PLATO's funding. Nobody on PLATO had ever experienced an on-line community before, so there was a lot of fumbling in the dark as social norms were established.

Over the years, PLATO has affected many lives in profound ways. So many real-life marriages have resulted from on-line encounters that such stories no longer seem remarkable. Usage Statistics

The CERL PLATO system logged 10 million hours of use between September, 1978 and May, 1985 (a period for which the most complete statistics are available).

About 3.35 million of those hours (over one third) were spent in Notes. About 3.3 million messages were posted. By the end of this period there were about 2000 notesfiles.

No figures are available for time spent in Personal Notes, "term-talk", or Talkomatic. But some numbers are known for games. Avatar alone accounted for about 600,000 hours, and Empire claimed another 300,000 or so. All told, games probably accounted for about 20% of PLATO usage during this period.

Few statistics are available for the many Control Data systems, but none were as large as the CERL system. An educated guess is that CERL accounted for about 25% of all PLATO usage worldwide.

The numbers are incomplete, but it is probable that people interacting with other people represented at least half of all PLATO usage. This is remarkable considering that the designers of PLATO never envisioned that communication between people would play more than an incidental role.

The PLATO Diaspora

Control Data ran into serious trouble in the late 1980's, and sold or closed many of its businesses. At the same time, microcomputers were becoming a more cost-effective platform for education than PLATO with its mainframe-based architecture, and many of the Control Data systems were shut down.

Today the PLATO name is owned by Minneapolis-based TRO, Inc., but TRO no longer runs any mainframe PLATO systems. Control Data's PLATO has been renamed CYBIS. Control Data Systems runs one CYBIS system itself and supports about a dozen others at university and government sites. There might still be some former Control Data customers running PLATO on their own.

At the University of Illinois, where it all began, PLATO has been renamed NovaNET. The U of I system racks up about 1.5 million hours of use per year, and is now operated by a private company, University Communications, Inc., of Tucson, Arizona.

The CYBIS systems still use the original PLATO Notes software. On NovaNET, a team headed by Dale Sinder rewrote Notes in 1991. Among the new features are multi-page notes and better search capabilities. But all of the key features of PLATO Notes, including the star structure of its notesfiles, have been kept.

Personal Notes has also been replaced on NovaNET. The new version uses a star structure to provide a level of organization that was never possible before. Each user's mailbox now looks

and works much like a group notesfile, with the user as its director and write-only access for everyone else. The new Personal Notes also sends and receives Internet e-mail.

PLATO's Progeny

As an educational/multimedia system, PLATO has many offspring. Its most successful direct descendant is TenCORE, a DOS-based authoring system. Macromedia's Authorware, an authoring system for the Macintosh and Windows, is also firmly rooted in PLATO.

As a communication system, PLATO has numerous other descendants. Many people who experienced the on-line PLATO community were inspired to replicate it on other platforms.

Lotus Notes is the best-known example. It was developed by Ray Ozzie, Tim Halvorsen, and Len Kawell, all of whom had worked at CERL in the late 1970's. It would be an exaggeration to call Lotus Notes a clone of PLATO Notes, because Ozzie expanded the concept to include powerful capabilities that were never contemplated for PLATO. But many of its basic features were modeled after PLATO Notes.

Here are a few other descendants of PLATO Notes:

- DEC Notes (previously called VAX Notes), a product of Digital Equipment Corporation originally written by Len Kawell. It is widely used on DEC's EASNet and on Starlink.
- Notesfiles, a public domain UNIX version of Notes written by Ray Essick and Rob Kolstad. In the early 1980's, it contributed significantly to the rise of USENET. Though eventually eclipsed by the News software, it is still used at some sites as a news reader or for local conferencing. Notesfiles can be obtained on the Internet at hilltop.bsdi.com:/pub/notesfiles.Z.
- COCONET's "Discussion" feature. COCONET is a UNIX-based software platform for running interactive multimedia on-line services, written by Brian Dear and largely modeled after PLATO. It is a product of Coconut Computing, Inc., of San Diego.
- Notefile, a Notes clone written in ALGOL for the Burroughs B6700 by John Eisenberg at the University of Delaware.
- FORA, a multi-user chat and messaging system for DOS written by Jim Bowery.
- The Connection, a XENIX-based BBS program written by Greg Corson.

Computer conferencing is just now hitting the big time, not only with Lotus Notes, but with large consumer-oriented services like America Online and Prodigy, and more sure to follow.

Among the on-line services I have seen, the WELL has best succeeded in building a community comparable to PLATO's. Ironically, the WELL has its roots with EIES and Confer; as far as I know, its founders were unaware of PLATO.

But the WELL is an intentional community. PLATO was an accidental one which emerged spontaneously in an environment that had been created for other purposes. In 1970 few suspected that a human community could grow and thrive within the electronic circuitry of a computer. PLATO demonstrated that this is not only possible, but inevitable.

Acknowledgments

Many thanks to Al Avner, a veritable fount of statistics. Additional information was provided by Rick Blomme, Jim Bowery, Greg Corson, Brian Dear, Sherwin Gooch, Mark Goodrich, Rob Kolstad, Dave LePage, Kim Mast, John Matheny, Dale Sinder, Joe Sneddon, Dan Tripp, and John S. Quarterman's book, *The Matrix*. Thanks also to John Quarterman for encouraging me to write this article.

The God of Backup

by Greg Rose

<ggr@acci.com.au>

There is a God whose prerogatives include backups. I don't know this God's name, but I know He (or She, or perhaps It, after all I wouldn't want to get sexist when talking about this God...) is definitely a vengeful God. Vengeful, angry, full of wrath, innovative in devising punishments and with a warped sense of humor. Altogether, He (She, It) is not a fun Guy (Gal, Goo) to have around.

Take a case in point. This is a true story. I know the names of the individuals and companies involved, but I'm not going to tell you. Anyway, there is this computer vendor, let's call them Vendor Inc., who sold a big, expensive computer to The Customer Corporation (say). This computer runs Customer Corp.'s absolutely vital, not to mention huge, corporate-everything database.

This system and database had been installed for over ten years, and the Data Processing department of Customer Corp. had been doing its job quietly and happily for that time. Every six months, in addition to Customer Corp.'s regular backup schedule, they took a full-system image dump, ending up with more than 10 tapes. Then, being the professionals that they are, Customer Corp.'s Data Processing people sent the tapes to Vendor Inc. and paid a lot of money to have them loaded and verified. Vendor Inc. actually had to assemble a big enough machine and restore and run the system on it. Then the tapes were locked into a fireproof safe at Vendor Inc.'s headquarters in case they were needed.

Well, late in December, Customer Corp.'s mainframe lost a cabinet full of drives to a small fire. "No Worries," said the D.P. Manager, a Mr. Lamb. "We'll just get new drives under maintenance and reload from the backups!"

The replacement drives were installed within a day; after all, Customer Corp. was big and important, and they paid a lot for maintenance too. The tapes were retrieved from Vendor Inc.'s safe, and the reload commenced. As it happened, the major backup had been done in November, only a couple of weeks earlier.

Tape 7 was the problem. Tape 7 couldn't be read. To be slightly more precise, what was there could be read, but there wasn't anything there. Nothing at all. It had been a brand new tape before the

backup was done, of course, no expense was spared, and it had all the characteristics of a brand new tape now. It looked like some sort of procedural error had occurred, and one of the other tapes had been written twice, or some such irrelevant thing.

It was, of course, recrimination time. Poor Mr. Lamb was called up before the Board of Customer Corp., and asked to explain why the system had been down for two weeks. Fortunately, he was able to point the finger: Vendor Inc. was supposed to ensure that the backups were all there. The CEO of Customer Corp. called the CEO of Vendor Inc. and after the conversation had settled down a bit, asked why the backups had been useless after customer had spent the last 13 years paying (a lot) for them to be verified? "I'll get back to you," said Vendor Inc.'s CEO, with a sort of quaver in his voice.

Vendor Inc.'s CEO was noticeably more confident when he called back. "I have some bad news for you — I hope you're sitting down," he said to Customer Corp.'s Head Honcho. "When the tapes came last month there was a note attached. It said that you were fed up with paying so much to have the backups verified, after all they were never used, and you just wanted us to store them. It was signed by a Mr. B. Counter. Of course we did as requested. I'm sorry."

Mr. Lamb was fired for failing to verify that the verification had happened. Somewhere, I'm sure, there was the sound of Hysterical Laughter.

Many people think that "backup" is a noun, an object that you refer back to when you need to. Others think that "backup" is a verb: the act of copying data. Any noun can be used as a verb. That's close, but here is the Truth.

A backup is a religious rite which propitiates the God of Backups.

It's obvious really. Everyone knows that if you take a good backup it will never be used. Even Murphy knew that. (Archives get used a lot. They are not backups. If you ever let your users know how easily you can restore files, they will delete important ones whenever they need temporary storage, then just ask you for them back. That's an archive, not a backup.)

The converse is also true. If you forget to take a backup, it isn't so bad, the God has other people

he can Hassle, for a while anyway. But if you take a backup onto bad media, or have a power outage in the middle of one, or take a backup with a script that must be run in "/" when your current directory is "/tmp", or any of these sorts of things, well, you asked for it. KerPow! The God of Backups wants his rituals done right, Or Else.

Just to establish my credentials (and show you how old I am), I'll give you another example. We'd been running Version 5 UNIX (no, that is not UNIX System V) for about six months when Version 6 came in. Doing a disk-to-disk copy of an RK05 using the block device took about 15 minutes. But V6 introduced raw devices!

We had three drives, not surprisingly numbered 0, 1, and 2. No tape, but removable packs. All backups were disk-to-disk. Well, we went to single user mode, put the system backup pack in drive 1, and started to `dd` from `/dev/rk0` to `/dev/rk1`. (I warned you about nouns and verbs, didn't I?) About halfway, our administrator stopped the copying, and said, "I'll just use the raw devices to make it faster." Of course, the moment the copying was interrupted, the God's beeper went off, and He began watching in fascination.

Our administrator quickly `mknoded` three times to create the raw devices. When the `dd` was

restarted, this time on the raw devices, the lights on drives 0 and 2 (!) came on, and stayed on for a satisfying amount of time as entire tracks were copied. It took about thirty seconds before we noticed that the wrong drive was lighting up. The DecWriter console still showed the typo that had mixed up the drive minor numbers, but it was too late. The system disk backup contained the beginning of the current system disk, but the end of the old one. The end of the actual system disk was intact, but the beginning of it was clobbered by the front of the user disk. The situation might have been recovered at this point, since the whole system disk image existed, but in two pieces.

Of course, since the system was running off that system disk, it promptly crashed. Maniacal Laughter.

So when you are performing a backup, remember the proper sense of gravity in the situation. Your backup ceremony should be systematic. Label the tapes neatly, immediately after they come out of the drive. Store them properly. Don't interrupt a backup, and don't let anything stop you from finishing one. Establish a schedule of Worship, and stick to it.

And don't ever start giggling during a backup, or tell backup jokes...

What's New

EDUPAGE

<edupage@ivory.educom.edu>

[Editor's Note: As part of my ongoing quest to assemble quotes and short reviews of periodical literature, I approached John Gehl <gehl@ivory.educom.edu> at EDUCOM about re-printing his twice-weekly blurb that summarizes information technology items. He generously granted permission! Edupage is a twice-weekly summary of news items on information technology provided as a service by EDUCOM - a consortium of leading colleges and universities seeking to transform education through the use of information technology. Let me know if you like this or not. <kolstad@usenix.org>]

You can subscribe to the full, long-version printed offering of EDUCOM REVIEW. For individual first-time subscribers, a one year trial subscription to EDUCOM REVIEW is \$18 (a 70% savings from the regular subscription price). Send check for \$18 to EDUCOM Review, 1112 16th NW, Washington, DC 20036. Offer good only in US and Canada.

Electronic music lawsuit. The National Music Publisher's Association is suing Compuserve for allegedly distributing "Unchained Melody" without permission. It's estimated the ballad has been infringed at least 690 times by subscribers who download the song onto their own computers. (Wall Street Journal 12/16/93 B1)

NSF extends network contract. NSF has extended a contract with Merit Inc. for managing the NSFNET backbone. The extension, worth as much as \$15 million, gives NSF until April 1995... (Chronicle of Higher Education 12/8/93 A17)

Obit. Apple Computer quietly stopped selling the Apple II last month. (Wall Street Journal 12/16/93 B1)

Intel predicts lower PC prices. A senior vice president at Intel predicts lower prices on PCs next year due to a doubling of chip performance over the next 12 months. Low-cost Pentium machines that now sell for about \$3,000 likely will cost about \$2,000 next year. (Wall Street Journal 12/15/93 B3)

U.S. chips tops. U.S. chipmakers controlled 41.9% of the world semiconductor market in 1993, compared to 41.4% for Japanese makers. (*Investor's Business Daily* 12/15/93 p.5)

Cable modems. Intel and General Instrument Corp. demonstrated a prototype cable modem that moves data at 10 million bits a second. (*Investor's Business Daily* 12/15/93 p.4.)

Wanted: Master's In Computer Science. A Northwestern University survey of 264 mid- to large-size companies indicates college grads with a master's degree in computer science will be among the luckiest job-seekers in 1994 -- demand will rise by approximately 22%. (*Tampa Tribune* 12/17/93 B&F8)

IBM Top Seller Of PCs. Preliminary year-end figures from International Data Corp. indicate that IBM will be the top seller of PCs this year, squeaking past Apple, with a 14% market share to Apple's 13.9%. This was a big year for the 10 largest companies, which accounted for 9.5 million of the 14.8 million PCs sold in the U.S. in 1993. (*Tampa Tribune* 12/19/93 B&F3)

Multimedia Patent Questioned. The Patent and Trademark Office will re-examine a patent granted to Compton's New Media last summer that covers a broad range of computer-based search and retrieval of graphics, text and video. The company claimed its patent covered basic technologies that could force competitors to pay millions of dollars in royalties. (*Wall Street Journal* 12/20/93 B5)

On-line Mags. Almost 100 magazines have signed up with on-line services in the last 6 months, without any adverse on their newsstand and subscription sales. The on-line services seem to inspire "talking" rather than reading, as readers chat among themselves and with writers and editors. (*New York Times* 12/20/93 C6)

Patent Titles. A free weekly e-mail distribution now gives all of the patents listed in the most recent issue of the USPTO Patent Gazette). For info: <patents@world.std.com>.

Hotel Messaging System. A new in-room messaging system ties into a hotel's telephone system to offer in-room faxing plus the ability to send and receive voice messages, get news, weather and sports info, and a copy of the hotel bill. (*Atlanta Constitution* 12/23/93 F2)

Wireless For All. AT&T pledge to make wireless communications services available to California minority, low-income, inner-city and disabled consumers; in view of that pledge, various con-

sumer groups will back AT&T's plans for a \$12.6 billion purchase of McCaw Cellular. (*Wall Street Journal* 12/23/93 B2)

Wishful Thinking. A Roper survey sponsored by IBM found that more than half of the respondents don't want a computer that requires a manual to use it. Two-thirds requested a computer that would recognize a user's face and automatically pull up the file s/he typically uses. (*Washington Post* 12/27/93 Business p. 13).

Too Much, Too Soon? Consumers are resisting the relentless push for increasing computing power and other enhancements that make PCs obsolescent. One consequence of consumer resistance is that about half the consumers who are buying PC's already own one, meaning growth has slowed tremendously. (*Atlanta Journal & Constitution* 12/25/93 D1)

Radio Shack In Russia. The first Radio Shack store in Russia opens this week on Leninsky Prospect in Moscow. (*Atlanta Constitution* 12/28/93 F3)

Software From India. India has become one of the world leaders in computer software production and Bangalore has become the site of India's version of Silicon Valley, with more than 100 computer software companies and tens of thousands of computer engineers. The work is often performed at state-of-the-art level, such as that being done on the Motorola's hand-held satellite system called Iridium. (*New York Times* 12/29/93 A1)

Outlook Good. In its "U.S. Industrial Outlook 1994" the Commerce Department identified information technology as 1994's most important growth area, estimating that satellite commerce will rise 22.6%, data processing 15.5%, electronic information 14.4%, and computer professional services 9.6%. (*Wall Street Journal* 12/29/93 A2)

Software Protection. The Patent and Trademark Office will hold hearings Jan. 26-27 in San Jose, Calif., on use of the current patent system to protect software-related inventions. The agency will also take comments Feb. 10-11 in Arlington, Va., on improving the patent examination process for protecting visual software through design patents. (*BNA Daily Report for Executives* 12/28/93 A4)

Apple's On-line Village. Apple is planning to offer a new on-line information service called *eWorld*, which will use a village as its metaphor for information services (with individual buildings in the village designating particular categories of information, such as business news, entertainment, etc.). (*Atlanta Constitution* 1/4/94 D5)

Dragnet. The FBI has posted a message on the Internet, asking for help in solving the Unabom case, which involves a series of bombings targeting the computer industry, the aircraft and airline industries, and universities. A \$1 million reward is offered for information leading to the conviction of the perpetrator. The FBI used a NASA computer to post the documents, which are accessible at naic.nasa.gov. (*Wall Street Journal* 12/31/93 p.10)

High-tech Work Spawns Decline In Suit Sales. The information age has had an unexpected effect on the men's clothing industry – suit sales have plummeted as men increasingly “dress down” for the office. (*St. Petersburg Times* 1/3/94 p.19)

The Well Is Sold. The Well (an acronym for Whole Earth Electronic Link) has been bought by businessman Bruce Katz, who plans to run it as a for-profit business and to upgrade the service's technology; the Well's general manager said, “We're interested in making this place a richer place for our users, but I don't see on-line shopping as part of that.” (*New York Times* 1/4/94 C3)

Intro To Computing. Video games – with their warrior and adventurer heroes – seem to be designed mainly for boys rather than girls. “That's very disturbing,” says media critic Marsha Kinder, “because video games provide an entry into the world of computers.” (*Atlanta Journal-Constitution* 1/1/94 D1)

Friend Or Foe? Technology like General Magic's Telescript, a new computer language that will allow you to send preprogrammed “agents” onto the networks to shop for best buys, make and modify your airline reservations, and perform many other services for you, has some people worried; their fear is that the agents could run amok and create the digital-network equivalent of an epidemic. (*New York Times* 1/6/94 C1)

Family PC Magazine. Walt Disney Co. and Ziff-Davis Publishing Co. will publish a new magazine called Family PC. The publication will target the 15 million or so households with PCs and kids. (*Wall Street Journal* 1/5/94 B7)

Mobile Copiers. QuadMark Ltd. is introducing a new portable copier the size of an egg carton and weighing less than four pounds. (*Investor's Business Daily* 1/5/94 p.6)

Business Card Scanners. The market for business-card readers that scan card data into PCs will soar to 200,000 units in 1997, up from 4,000 last year according to BIS Strategic Decisions. (*Wall Street Journal* 1/6/94 A1)

Why Do It? “51 Reasons To Build The National Information Infrastructure” is a selection of stories collected by FARNET to help answer that question; it's available electronically by ftp, gopher and BRS/Search at cni.org.

Highway Referee. Ken Auletta says in the *New Yorker* that four basic issues dominate the Washington debate about the electronic superhighway: antitrust policy, universal access, fairness, and privacy, and that “despite all the talks of deals and converging technologies, and of how seven players – cable, telephone, computer, studios, broadcasting, publishing and consumer electronics – may shift allegiances, perhaps the role most often overlooked has been that played by an eighth force – the suddenly assertive government referee.” (*New Yorker* 1/17/94 p.49)

DBS & Cable Competition. The kind of cable system that can deliver 500 channels is at least several years away, but Direct Broadcast Satellite (DBS) television, which can send digital signals to homes equipped with an 16” dish and set-top decoder box, will be available in a few months. By being first with a lot of programming, DBS will offer healthy competition to cable systems. (*U.S. News & World Report* 1/17/94 p.56)

Techno-savvies Are Hot. Madison Avenue is lusting after its newest-discovered consumer species – the Techno-Savvy. The average T-S is male; 42 years old; married; and lives in a suburb of Los Angeles, New York or San Jose. Forty-eight percent use a computer every day and 54% earn more than \$100,000 a year. (*Wall Street Journal* 1/11/94 B1)

Bell Atlantic Pledges Internet Connections. Bell Atlantic and TCI announced plans yesterday to link 26,000 K-12 schools to the national information infrastructure. The Basic Education Connection, as the program is called, initially will include free installation and linkages to help schools reach databases such as the Internet, as well as free educational cable TV programming and access to certain data services. The companies are soliciting advice from Internet users on how best to coordinate efforts, and suggestions may be sent to ba.tcieduc@tcinc.com. (*Washington Post* 1/11/94 D4)

Intel To Develop Conferencing Standard. Intel has assembled a group of companies with the goal of coming up with an industry standard for desktop video and document conferencing. (*Wall Street Journal* 1/11/94 B3)

CDs Outsell Cassettes. For the first time since they were introduced, compact disks outsold cassettes last year. (*Wall Street Journal* 1/7/94 B2)

Measure Of Innovation. IBM won the most U.S. patents in 1993, and Eastman Kodak was fourth on the list. Japanese companies hold six of the top 10 patents. (*The Trenton Times* 1/13/94 D1)

Nii Update. The Clinton administration's Information Infrastructure Task Force has set up a Gopher server, allowing Internet users to retrieve documents, speeches and other information related to the proposed information superhighway. Use the telnet or Gopher function to reach iitf.doc.gov and logon with the password "gopher." (*Chronicle of Higher Education* 1/19/94 A25)

Super Phones. Forget the black box—AT&T is betting on the ubiquitous telephone emerging as the gizmo of choice for accessing the information superhighway. AT&T's Project Sage is working on a phone that can be the central controller for routing information to and from household electronic gadgets such as TVs, VCRs, PCs, fax machines and video cameras. (*Business Week* 1/24/94 p.37)

Cray Crunching. Cray Research computed the largest prime number known, with 258,716 digits. (*Business Week* 1/24/94 p.44)

Faulty Software Proves Legal Bonanza. A new legal niche is emerging for clients seeking damages due to malfunctioning computer systems. "In the 21st century, computer performance is going to play a role in most litigation," notes an Association of Trial Lawyers of America member. (*Wall Street Journal* 1/21/94 B1)

New Compression Technology. An "upstart" company called Digital Compression Technology boasts its new technology will enable 10 channels of video to be sent over existing copper telephone wires. (*Wall Street Journal* 1/27/94 B8)

Really Smart Chips. Researchers are experimenting with electronic microchips that use living brain cells. The embryonic cells are placed on silicon or glass chips and induced to grow along desired paths. (*Wall Street Journal* 4/1/94 B7)

Intel's Next Chip. Intel will demonstrate its next-generation P6 chip this time next year, according to its CEO. (*Wall Street Journal* 1/28/94 B5)

Crayons To Computers. Micrografx is teaming up with the maker of Crayola crayons to develop a line of software programs for kids. (*Wall Street Journal* 2/1/94 B6)

Sex Titles Driven Off Info Highway. Canada's best-known computer science school, the University of Waterloo, banned from its campus five Internet [newsgroups] dealing with violent sex and other areas out of concern that the contents contravene laws on pornography and obscenity. A university spokesperson says that while Waterloo tries to protect freedom of expression as much as possible, it must draw the line when it goes beyond the Criminal Code. (*Toronto Globe & Mail* 02/05/94 A1).

Britannica Online. The Encyclopedia Britannica will be available over the Internet to universities and some public libraries. The online version includes hyperlinks making it possible for an article to link to related articles and illustrations. The NCSA-developed Mosaic software program will be used for searching the online encyclopedia. (*New York Times* 2/8/94 C1)

Internet Clearinghouse. A clearinghouse on network resources has been set up by a University of Michigan graduate student. The Clearinghouse for Subject-Oriented Internet Resource Guide provides access to a group of network guides to more information, from anthropology to theology. The Clearinghouse can be reached through a variety of Internet tools: Gopher users should connect to gopher.lib.umich.edu and look under "What's New and Featured Resources," and then under "Clearinghouse." (*Chronicle of Higher Education* 2/9/94 A26)

Encryption. The Clinton Administration is adopting the "Clipper Chip" an encryption device with a "back door" that allows law enforcement agencies with court orders to eavesdrop on digital communications. The decision has been denounced by industry groups as well as civil liberties groups concerned about privacy. (*New York Times* 2/5/94 A1)... The CPSPR organization is organizing a protest, which can be joined by sending e-mail to clipper.petition@cpsr.org with the message "I Oppose Clipper." Ftp/gopher/wais to cpsr.org /cpsr/privacy/crypto/clipper/ for info.

Internet Break-ins. The Computer Emergency Response Team (CERT) has reported a large number of break-ins to computers on the Internet and has urged computer users and systems administrators to change passwords and increase security precautions. (*Washington Post* 2/4/94 A1)

News from EurOpen National Groups

GURU (Romania UNIX User Group)

About GURU activities (taken from a recent letter from GURU to EurOpen)

by Alexandru Rotaru

In January 1994, GURU started a training program, together with SOROS Foundation from Bucharest. The program is dedicated to the Romanian high schools, and we have started with 26 of them from different towns in Romania. The training has been structured under four levels (courses), each of which is two weeks long. These levels are as follows:

- a) Basic UNIX training and dial-up email administration. After this they get one PC with SCO UNIX OS and 1 modem for dial-up email for their own usage at their location. They also receive 4 PC based LAN workstations with PC-NFS and X/Windows client software. One workstation is for Linux. At the end of this training module, they also get a complete Linux Slackware (the last release) package plus the printed documentation for Linux (about 800 pp. of text). All of this is financially supported by the SOROS Foundation. They are also tested and certified by GURU, and they continue to work with us by email.
- b) UNIX programming. It has two parts. One is dedicated for basic UNIX C programming and POSIX compliance for programmers. The other is for MOTIF programming under X/Windows.
- c) Networking and application development on NFS based LANs. Some IP administration topics for the "full" Internet connectivity are also presented.
- d) Special training with some modern tools: GNU compilers, Emacs, Ocean & Nelsis, PVM, ARJUNA, PERL and some AI & Neural Net tools. This training is performed on Linux by the people from GURU who used these applications and have the expertise to train the people in this area.

This training is being done to start some modern activities at the high school level in Romania and to select "youngsters" for future university training. This is the first national training program in Romania in which we are involved as a group.

The main tool for making these high schools operational is email. We are trying to help them build the "high schools' net" with several value added services like news, and list servers e.o for information interchange, and for starting their own working groups inside the country in this way.

GURU once again received from X/Open, the task for Xtra Survey this year. This time we proposed a 100 sites questionnaire in Romania.

The composition of the group it is the same: 30% of our members are UNIX high-level staff with a lot of experience and the remainder are users of Open Systems.

We worked on our internal email structure, and now all the board from GURU and some others have email addresses.

The general correspondence address for guru is <guru@guru.ro>. The postal address for GURU is: P.O BOX 63-42, Bucharest, Romania.

Some events during this year are:

- ROSE'9 – Romanian Open Systems Event – International Conference & Exhibition. This year it will be held in Bucharest in November. The date is not settled yet because of arrangements proposed by EurOpen for inviting speakers at ROSE'94.
- Each month we have a meeting with our members in which companies are invited to make presentations. In March we have invited Novell to present the architecture and the internals of UNIXWARE.
- We'll start in the near future our own UNIX & Open Systems magazine.

Other "computer area" events in Romania:

- IFABO Bucharest, 1-5 March 1994. Most of the companies involved in informatic systems' business for Romania will participate. We have a special meeting at GURU with some of the participants.
- CERF '94, Bucharest, 9-15 May 1994, computer fair organized by COMTEK International. We have also a meeting at GURU with some companies joining CERF '94

An Update on UNIX-related Standards Activities

by Nicholas M. Stoughton
USENIX Standards Report Editor
<nick@usenix.org>

Report on the IEEE Standards Board

Mary-Lynne Nielsen <m.nielsen@ieee.org>
reports on the September 1993 meeting:

This was a very busy meeting, with a lot of major actions for the Portable Applications Standards Committee (PASC, the sponsor of, among other things, the POSIX projects): document approvals, PAR (Project Authorization Request) approvals, new review policies, and a whole new numbering scheme for the POSIX documents.

More on renumbering later, but let's get the easy stuff over with first.

SPAsystem Update

Work on the SPAsystem proceeds apace, and the IEEE Standards Board is preparing to create an Industry Advisory Group (IAG) to assist and advise in this process. Preliminary interviews have been conducted with potential candidates, and the Board is looking at creating this committee in 1994. Just as a reminder, the SPAsystem will be the springboard for all the future electronic development and delivery of standards products. The first part of this, the IEEE bulletin board, is already running and accessible via modem; Internet access is the next big step for this project. In addition, the IEEE is working on developing standards on-line in tandem with moving towards developing standards in SGML, the Standard Generalized Markup Language. This step is in research, with an aim to mount pilot projects in 1994. Previous snitches have gone into some detail concerning the development of this project.

Metric, Metric Everywhere

Over the past year, all the Boards of the IEEE received presentations from members of the IEEE Metric Policy committee concerning adoption of a broad-based Institute policy on the use of the metric system and SI units. This policy was approved by the Standards Board and was voted on by the IEEE Board of Directors (the governing Board of the entire Institute) in November. At that time, this policy was approved. It basically states that the IEEE shall actively support the use of the International System of Units (SI), both by educating its members as to its use and through

active implementation of the system in its practice. For standards, this means that SI units are the preferred method for unit symbols. While this doesn't begin to resolve the issue of whether a kilobyte is 1000 bytes or 1024 bytes, it is a step in the right direction.

P1003.4

One of the more eagerly awaited standards, IEEE P1003.4, was approved at this meeting by the Review Standards Committee (RevCom). It will be published in the spring of 1994 as IEEE Std 1003.1b. Don't worry about the numbering, I'll explain that later.

NesCom Actions

NesCom, the New Standards Committee, discussed at length the need for good cross-communication between itself and Accredited Standards Committee (ASC) X3, an independent standards group that also covers the field of information technology, but whose membership is company-based rather than individual-based. There have been problems with overlapping scopes of projects in these two groups in the past, particularly when projects are proposed to the American National Standards Institute (ANSI) Board of Standards Review (BSR), which is the US coordinating body for project approvals. In an effort to educate and minimize overlap, NesCom moved to try and avoid those problems at the ANSI level by having NesCom review ASC X3 new project proposals (which are called SD3s) and include X3 for information on all NesCom mailings, including PAR distribution.

A great deal of activity occurred in relation to PASC PARs. Five new PARs were approved. Two of these were a revision and splitting of a previous PAR (IEEE P1003.6) to show the security revisions to IEEE P1003.1 and IEEE P1003.2. One PAR was for a revision to IEEE P1003.5, and one was a new PAR. Also, the PAR for the language-independent (LI) version of IEEE P1003.1 was revised and renumbered.

In addition, three PARs were withdrawn at this meeting. One, IEEE P1003.19, was withdrawn because the working group was chartered to develop Fortran 90 bindings to POSIX.1, and that field is not mature enough to encourage standardization. Two others, IEEE P1003.16 and IEEE P1003.16a, were withdrawn due to the death of

the LIS requirement in PASC and hence the removal of the need to have a separate C language binding to POSIX.1. Details of these PARs are given below.

Numbers? What about numbers?

All right, take a deep breath! Here we go!

Yes, many of the POSIX projects have been renumbered by NesCom. Before I get into the whys and wherefores, let's try to understand some of the history and get the details over with.

The POSIX series of standards, referred to by their base number of IEEE P1003, have been under development for a number of years. Originally, the project was relatively small in size; a few key standards existed, along with some adjunct standards projects in development. However, over the years, the growth of work this area has been exponential. There are now over 30 projects that have been or are being developed that fall under the banner of "POSIX."

The numbering for these standards has been varied, in part to show their relationships to base standards, to the working groups developing those projects, or to other projects or supplements in the series. In other words, we've had all kinds of different numbering for POSIX. Both base standards and amendments have been numbered 1003.x; amendments have been numbered both 1003.x and 1003.n (for an alphabetic designator); and projects have been numbered with a "double-dot" number to show their relationship to one of these "base" single-dot numbers (such as IEEE P1003.7.1).

In addition, the POSIX series of standards have been adopted or proposed for adoption as international standards. The international committee in charge of overseeing the POSIX international development, ISO/IEC JTC1 WG15, recommended a certain structure for this proposed work. That structure consisted of one number for the whole series, 9945. (This number parallels the IEEE equivalent of 1003.) However, ISO/IEC proposed that only three parts exist in 9945. All other projects would have to fit into that three-part plan as supplements to those "base" standards.

Remember, at this time, there were already about 10 PARs in existence in PASC. So when ISO/IEC decided that there would only be three parts, the system was already broken. What did have to change, however, was the content of the documents themselves; the IEEE POSIX projects had to make themselves fit into this structure by mapping themselves, section by section, to the content of the "base" standard they were amending.

This process has been a long, arduous, and ongoing one in PASC, as many standards needed to be entirely restructured to show their relationship to what were now the three base standards. However, the original numbers assigned to these projects by the IEEE had never been changed to reflect these evolving relationships. People learned that IEEE P1003.4 was going to be part of IEEE P1003.1 and left it at that. All that has begun to be adjusted by action of the NesCom at this meeting.

So what prompted any change? Basically, the PARs to revise IEEE P1003.6. These new PARs split the old POSIX.6 work into two parts, one that amended IEEE P1003.1 and one that amended IEEE P1003.2. The proposed PARs used the double-dot numbering scheme (P1003.1.6 and P1003.2.6). The chair of NesCom would not accept these proposed numbers because they didn't meet the current numbering system used by IEEE standards. (When this numbering system was approved, POSIX was considered and recognized as an exception to it.) This led to a detailed discussion of the overall POSIX project numbering, the end result of which was a conclusion from NesCom that the existing POSIX numbering scheme needed to be cleaned up and corrected.

Most of the discussions concerning this occurred once the NesCom agenda (with PARs) was received by the committee for their 40-day review period prior to the meeting. Thus, Jim Isaak and I (Jim is also a member of NesCom as well as PASC chair) were able to discuss this at great length with the committee members. Despite giving many reasons for why the old numbers should be kept, NesCom remained adamant about bringing these PARs in line. This then left the problem of creating another layer of broken numbers on top of an already broken numbering system. Finally, it was proposed that all of the current POSIX projects not yet published be renumbered to match the existing standards numbering scheme. Period. In toto.

This task was partially accomplished at the meetings of NesCom; the easiest projects to renumber were handled. There are some projects in which further clarification from the committee is needed prior to renumbering, and these problems will be resolved for the December meeting. For instance, it was unclear whether 1003.7 should keep a 1003 number because of a proposal to change its ISO/IEC number from 9945-3 to an entirely new number apart from the 9945 series. So the numbering on that will be revised later. In addition, there are some projects that have not yet been renumbered because their early stage of development does not yet clearly pinpoint to which base standard they will belong.

In other words, most of the POSIX family of standards, and all of the mature projects, have been or are going to be renumbered to fit into the current acceptable numbering scheme of NesCom. In some cases, this means no change at all to the PAR. In others, it means a major change.

Sure, this can and will be confusing. No question about it. But NesCom forced the issue, and it had to be dealt with. I suppose you could think that it should have been dealt with all those years ago when the ISO/IEC and PASC numbering schemes diverged, if you need a consoling thought. And there are a couple of advantages to these numbering changes. One is that all the POSIX numbers will be in one numbering scheme, and this numbering system clearly shows which documents are meant to be supplements to a base standard and which are base documents themselves. Previously, the numbers did not express this relationship.

One comment – why did this happen now, at this stage in development? You have to keep in mind that all committees of the IEEE Standards Board,

including the Board itself, are as dynamic as the working groups developing standards. The members change every year; some stay on for years, some stay for only one year. The tone and action of each committee are a direct reflection of the current membership. Because there was a new vice-president of the IEEE Standards Board this year (for the first time in three years), NesCom gained a new chair and a lot of different members. Basically, that's why the issue came to a head; it was perceived as a problem by the current membership and the chair.

It is expected that all PASC working groups will start using the new numbers, probably in conjunction with their old numbers, immediately. (In large part, the use of both numbers will occur to address the expected confusion.) Over time, the old numbers will be phased out and the new numbers used exclusively.

Here is a list of the current changes in IEEE POSIX numbering, along with the numbers that don't change because they don't fold into P1003.1 or P1003.2. There will be more to come in December.

Old number	New number	Description
P1003.0	Same	Guide to POSIX OSE
P1003.1	Same-Published	System Interface
P1003.1a	Same	System Interface (revision)
P1003/LIS	1372	Language Independent Specification
P1003.2	Same-Published	Shell and Utilities
P1003.2a	Same-Published	
P1003.2b	Same	
P1003.3	P2003-Published as IEEE Std 1003.3-1991	Test Methods
P1003.3.1	P2003.1-Published	
P1003.3.2	P2003.2	
P1003.4	P1003.1b-to be published	Realtime Extensions
P1003.4a	1003.1c	Threads Interface
P1003.4b	P1003.1d	
P1003.5	Same-Published	Ada Bindings
P1003.1.6	P1003.1e	Security extensions – System Interface
P1003.2.6	P1003.2c	Security extensions – Shell and Utilities
P1003.8	P1003.1f	Transparent File Access
P1003.9	Same-Published	FORTTRAN Bindings
P1003.10	Same	Supercomputing Profile
P1003.11	Withdrawn	Transaction Processing Profile
P1003.12	P1003.1g	Protocol Independent Network Interfaces
P1003.13	Same	Realtime Profile
P1003.14	Same	Multiprocessor Profile
P1003.15a	P1003.2d	Batch Extensions for Supercomputing
P1003.16	Withdrawn	C Language Binding to POSIX.1LIS
P1003.16a	Withdrawn	
P1003.17	P1224.2-Published	X.500 APIs
P1003.18	Same	Platform Environment Profile
P1003.19	Withdrawn	Fortran-90 Bindings
P1003.21	Same; to be revised	Realtime Distributed IPC
P1003.22	Same; to be revised	

New PARs

P1003.7.3 Standard for Information Technology-Portable Operating System Interface (POSIX)-Part 3: System Administration-Amendment: User Administration

NOTE: This approval was contingent upon the Sponsor Chair returning to NesCom in December with an "appropriate numbering system."

Revised PARs

P1003.1.6 Standard for Information Technology-Portable Operating System Interface (POSIX)-Part 1: System Application Program Interface (API)-Amendment n: Protection, Audit, and Control Interfaces [C Language]
NOTE: Renumbered to P1003.1e.

P1003.2.6 Standard for Information Technology-Portable Operating System Interface (POSIX)-Part 2: Shell and Utilities- Amendment n: Protection and Control Utilities
NOTE: Renumbered to P1003.2c.

P1372 Standard for Information Technology-Portable Operating System Interface (POSIX)-Part 1: System Application Program Interface (API)-[Language Independent]
NOTE: This PAR revised the PAR for P1003.1/LIS.

PARs for Revision of Existing Standards

P1003.5 Standard for Information Technology-POSIX Ada Language Interfaces-Part 1: Binding for System Application Program Interface (API)

Withdrawn PARs

P1003.16 Standard for Information Technology-POSIX C Language Interfaces-Part 1: Binding for System Application Program Interface (API)

P1003.16a Standard for Information Technology-POSIX C Language Interfaces-Part 1: Binding for System Application Program Interface (API)-Amendment 1: System API Extensions

P1003.19 Standard for Information Technology-POSIX Fortran 90 Language Interfaces-Part 1: Binding for System Application Program Interface (API)

Some Humor

I recently received the following from Dr. Kerry Raymond at the University of Queensland, and it may amuse those of you who have ever sat on a standards committee:

I'm On a Committee

Oh, give me your pity, I'm on a committee
Which means that from morning to night,
We attend and amend and contend and defend
Without a conclusion in sight.

We confer and concur, we defer and demur
And re-iterate all of our thoughts.
We revise the agenda with frequent addenda
And consider a load of reports.

We compose and propose, we suppose and oppose
And the points of procedure are fun!
But though various notions are brought up as motions
There's terribly little gets done.

We resolve and absolve, but never dissolve
Since it's out of the question for us.
What a shattering pity to end our committee
Where else could we make such a fuss?

Report on POSIX.0: Guide to POSIX OSE

Kevin Lewis <klewis@gucci.enet.dec.com> reports on the January 10-14, 1994 meeting in Irvine, CA:

From The Battlefield

Work to get POSIX.0 approved continues. The ballot recirculation is in. Let me first give you the statistics. There are 86 total people in the balloting group of which 81 are eligible to vote. A total of 75 ballots were returned. The breakdown of those votes are as follows: 45 affirmative, 17 negative, 13 abstentions. This represents a 92% return. The 45 votes represent a 72% affirmative. The ballots consist of 167 comments and 182 objections, which represent about 25% of the total submitted during the first round of balloting.

Before commencing resolution of the recirculation ballots, in Irvine, we discussed a letter that had been received by the IEEE from the ACM. This letter focused on the overall IEEE balloting process, the concern that some of IEEE work overlaps with standards work within X3, and that our guide document still lacks the necessary level of consensus. A portion of our rationale for rejecting specific objections was inadequate.

This letter amounted to a hand grenade lobbed into the middle of our work, or, to quote working group members, "a tactical nuclear attack." I won't go into fine detail here, but the group did meet with two people, who were wearing ACM hats, to share their concerns along with those of ACM. However, some working group members who were also ACM members were quite disturbed by the tone of the letter, part of which included an 'ad hominem' attack against the working group itself. They were also distressed by the

approach taken by ACM of sending such a letter to the IEEE without first having a dialogue with the group. Words such as 'protest' and 'malfeasance' made their way into the discussion.

In my humble opinion, the only part of the letter I considered valid (and I'm quite sure I would have the unanimous assent of the group on this) was that portion addressing our rationale. This, by the way, was quite helpful. It redirected our efforts for the better during the week. The group decided to return to the unresolved objections from the first ballot for the purpose of reviewing each one for possible acceptance or correcting/strengthening our rationale for rejection, which I admit was both weak in places and occasionally arrogant.

We completed this task, but did not get to the recirculation ballots. Because of this, and also due to the overall feeling in the group that more productive resolution work could be done at a meeting away from the quarterly PASC (Portable Applications Standards Committee) meetings, we agreed to schedule an additional meeting to take place in March in the Washington, D.C. area, specifically for the section leaders. The only activity at this meeting will be resolution of the recirculation ballots. The exact date has yet to be determined.

I feel quite strongly that we will be able to complete all of the recirculation ballots at this March meeting. What remains now is the review-and-comment action by SC22, the ISO subcommittee responsible for POSIX, which is now in progress. It looks like it will be October before we have a document ready for submission to the IEEE Standards Board.

One more thing: the POSIX.0 working group is scheduled to meet for two days at the April PASC meeting in Lake Tahoe. This will be a skeleton crew to effect coordination with and provide representation to some other key PASC committees, such as the Profile Steering Committee and the Sponsor Executive Committee. In addition, this crew will monitor the resolutions to the international committees that directly or indirectly affect the guide effort.

Report on POSIX.22: Computer Security Framework

Randall Wayne Simons <rsimons@somnet.sandia.gov> reports on the January 10-14, 1994 meeting in Irvine, CA:

The POSIX.22 committee is defining a framework for distributed computer security. The framework will be a common reference model to guide

members of other POSIX committees in addressing security needs in the standards they are defining.

At this first POSIX meeting I have attended, my main impression was of heads silently bowed over clacking keyboards as multiple laptops were simultaneously applied to modifying a document. David Rogers, chair of the committee, brought a troff version of the X/Open Snapshot called the "Distributed Security Framework," POSIX.22 wants to keep the X/Open and POSIX documents in sync since both groups are working on the same problem. The most recent version of the document had just been reviewed by X/Open, and there were numerous suggestions for improvement, including many that required some restructuring of the document. POSIX.22 took on this task, and simultaneously reviewed and added their own improvements. Different sections of the document were distributed to each committee member who then did the cutting, pasting, and merging.

The reorganized document begins by introducing top-level information system security concepts, terms and models. There is a description of threats, most of which were moved to an appendix. More detailed models define security architectures and characteristics of interfaces to security services. Finally, the individual services and interfaces are modeled and described in detail. Interfaces support both management and operational functions for each of the services.

The basic services included are: authentication, access control, security audit and cryptographic services. At a higher level, domain interaction services, which combine various basic services in a distributed environment, include user authentication and secure association service.

After more review and revision by both X/Open and POSIX.22, the Framework document will be ready for balloting around July. The balloting group should meet in April, so be watching for it. POSIX.22 had seven people attend this meeting. There was plenty of work to go around. Anyone willing to help develop the POSIX Computer Security Framework will be more than welcome at future meetings. There is much to be done in security for POSIX – see the report from POSIX.6.

Report on POSIX Test Methods

Fred Zlotnick <fred@mindcraft.com> reports on the January 10-14, 1994 meeting in Irvine, CA:

The requirement that POSIX working groups develop test methods in parallel with their standards was suspended at the April 1993 meeting,

and then finally withdrawn at the following July meeting. Nevertheless there are two active test methods activities and more in the works. The working groups, which met at the October POSIX meeting in Bethesda and at the January meeting in Irvine, are group 2003 (which is revising POSIX.3, the standard that describes how to write test methods) and group 2003.2 (which is developing test methods for POSIX.2, Shell and Utilities). Technically it wasn't the 2003.2 working group that met, but more about that later. Both of these groups are chaired by Lowell Johnson of Unisys.

Revision to POSIX.3

Working group 2003 has been writing a revision to POSIX.3 for about a year and a half. Although POSIX.3 has been used successfully to write test methods for POSIX.1, and its methodology has formed the basis for quite a few commercial test suites, the use of this methodology has revealed a number of problems. The purpose of this revision is to deal with these problems:

- It is difficult to use POSIX.3 to write test methods for standards that modify other standards. Real-time (POSIX.1b, which used to be POSIX.4) is a good example. Because the real-time standard consists of a collection of optional extensions to POSIX.1, every assertion for real-time must be conditional (type C or type D). But there are other conditions within many real-time assertions, and this makes the statement of each assertion in POSIX.3 format rather cumbersome. Moreover, some of the options of POSIX.1b place additional semantic requirements on POSIX.1 interfaces such as `open ()`. Writing the assertions to test these requirements raises questions not adequately addressed in POSIX.3-1991: How should they be numbered? How should they be conditioned? How should they be classified (assertion-typed)?
- A number of the users of POSIX.3 have found the standard difficult to understand. A number of related but distinct concepts in POSIX.3 have been confused by users of the standard.
- ISO had difficulty with the terminology of POSIX.3, which is not always consistent with that of other test methods standards at the international level.
- POSIX.3 was originally designed for and is specified as only applicable to POSIX standards. The IEEE's Portable Applications Standards Committee (PASC) currently manages a number of projects, some of which fall under the POSIX umbrella. Yet the test methods methodology of POSIX.3 applies to, and should be specified for,

other PASC working groups, such as P1327 and P1328. In general, the scope of POSIX.3 should be broadened.

Working group 2003 began the week in Irvine with Draft 2.0 of the revised standard. This draft had been completed by the group's technical editor, Anthony Cincotta of NIST, just prior to the meeting. By the end of the week the group had agreed on a set of changes that, when completed, will produce Draft 3. This draft should be suitable for mock-ballot.

The basic methodology of assertion-based testing has not changed in this revision, but the form in which assertions are written has changed drastically. The familiar, frequently misunderstood, and often vilified 2x2 matrix of assertion types is gone. The syntax of an assertion now closely resembles a conditional sentence, with possibly many nested conditions. If an assertion applies only when conformance to a particular version of a standard (e.g., POSIX.1-1993) is being tested, a condition indicates this fact. If an assertion depends on support of an option (e.g., job control) a condition indicates this fact. Sometimes an assertion may specify required behavior but may only be testable if the implementation supports optional features (such as certain appropriate privileges). If so, a condition indicates this fact. Assertions are now labeled by assertion IDs rather than assertion numbers; an assertion ID is a string.

The new assertion structure promises to make assertion writing easier and to allow the structure of test methods standards to more closely parallel the base standards against which they are written.

POSIX.2 Test Methods

For the last four meetings, the 2003.2 working group has been laboring over the ballot resolution for the ballot of Draft 8. According to IEEE rules, this means that it is not really the 2003.2 committee that is meeting. Ballot resolution is done not by the committee that wrote the draft, but by a group of technical reviewers. They just happen (in this case, as in most) to be many of the same people.

Ballot resolution for Draft 8 has been a slow job for a number of reasons. One is the size of the task: there are almost 9000 assertions in Draft 8. Another is that despite its size, Draft 8 is incomplete, and a number of ballot objections make note of this fact. Some of the missing pieces (like assertions for `yacc`) have not been easy to supply. Nevertheless, part of the task of resolving these objections will be to fill in those missing

pieces. Another problem is that participation in this working group has not been as consistent as one might like, although the October and January meetings were well-attended.

In addition to the incompleteness of Draft 8, major ballot issues include the fact that the test methods must be locale-dependent but the draft frequently addresses testing only in the POSIX locale. Moreover, Draft 8 is not explicit about this fact. Other problems include the omission of reasons for classifying assertions as extended, and the omission of clear references for reference assertions.

By the end of the October meeting, reviewers had made enough progress to enable the technical editor, Andrew Twigger of UniSoft, to produce interim Draft 8.5. This will not be balloted, but it has been useful as a working document at the January meeting. At that meeting the technical reviewers completed ballot resolution. The technical editor now has to integrate the resulting changes to produce Draft 9, which will go out to ballot.

Test Methods for POSIX.1b

At the Irvine meeting, the PASC Sponsor Executive Committee (SEC) approved a new Project Authorization Request (PAR) for test methods. The PAR creates a POSIX 2003.1b project under the direction of the 2003 working group. Its goal is to write test methods for POSIX.1b.

You may recall that during the “test methods wars” the POSIX.4 working group was grandfathered out of the requirement (since lifted) to develop test methods along with base standards. Thus there are no test methods, even in draft form, for POSIX.1b. Yet there is substantial interest in the development of conformance tests for POSIX Real Time, and such tests need a specification. In Irvine, a number of organizations, including representatives of several DoD agencies (DISA, JITC), were committed to provide resources to develop these test methods. Ken Thomas of JITC has offered to serve as vice-chair of the 2003 working group for the direction of the 2003.1b effort. Bruce Weiner of Mindcraft has offered to act as technical editor for the test methods document.

Report on POSIX.5: ADA Bindings

Delbert L. Swanson <DSWANSON@mhs.sp.para-max.com> reports on the January 10-14, 1994 meeting in Irvine, CA:

The primary charter of the POSIX.5 group is to produce Ada language bindings to POSIX stan-

dards. The Ada binding for POSIX.1, POSIX.5, has been published as an IEEE standard, and we are now preparing bindings to the Real-Time Extensions standards being developed by the POSIX.4 group. These bindings have been designated as POSIX.20.

Draft 2 was developed as a “thin” binding to the Real-Time Extensions. That is, it merely made a correlation between the constructs defined in the C version and the constructs in the Ada version. None of the explanations or semantics are repeated. This was done following what was the policy of IEEE and the ISO community – that all language bindings would be thin bindings of a normative language independent specification (LIS) of the standard. Actually, our approach was a compromise even then, since there was not yet a completed LIS version.

Circulated for a first ballot last summer, this draft was updated to account for comments and objections. In the meantime, the policy on thin bindings to LIS versions of standards changed, and so the group has been revising the document.

The next draft will be a thick binding – a complete specification of the interfaces for Ada applications. The advantage is that users will not need to refer to multiple documents (Ada and C) to understand the behavior of the Ada interfaces. The disadvantage is in the maintenance mode: if the baseline document changes, the binding document needs to change correspondingly. Moreover, it takes more work to produce a thick binding than a thin one.

We expect to work on more issues between meetings, and then polish the draft up to be ready for another full ballot after the April, 1994 meeting.

In January, the group re-examined our approach to Ada bindings to the threads extensions. We have concluded that almost all of the functions offered in POSIX.4a are going to be provided for Ada applications in the revision of the language commonly called Ada9X, which is expected to be granted standard status within the year. It seems beside the point for us to duplicate, as OS functions, these capabilities, which will soon be available as language constructs. A couple of remaining pieces will be incorporated into the new draft of POSIX.20.

The status of our coordination ballot on POSIX.4a, the threads extensions is an item of some concern to the group. All but a couple of our objections were resolved in discussion. Unfortunately, the objection that we consider most important has been rejected on the grounds that it would reduce

consensus. It is our view, particularly when handling signals, that it is important to be able to mask asynchronous signals for the entire process. This is important in Ada runtime environments, and it will also be important within C programs. The current C interface includes only per-thread signal masks. It is uncertain what the resolution of this issue will be.

Meanwhile, we are preparing a revision to POSIX.5, to correct errors found in the standard by implementers (missing parameter, missing function definition, error condition oversights). The only way to make "substantive" changes, even for errors, is to revise the standard, which means balloting, etc. Revisions should be ready for ballot as soon as the administrative details are taken care of.

Report on POSIX.6: Security Extensions

Lynne Ambuel <ambuel@dockmaster.mcsc.mil>
reports on the January 10-14, 1994 meeting in Irvine, CA:

Introduction

As a first time snitch, I would like to divulge my thoughts on standards – from a security geek's point of view. Subjects include the peculiarities of information security and those who live by it, the activities taking place, and the status of the POSIX security working group (previously known as P1300.6). Other issues may creep into the discussion, but everything will relate (no matter how obscurely) to these greater issues.

A Different Animal

Computer Security specialists are used to being called names like "different," "special," and even "strange." Although some might take offense, I must agree with the characterization. Computer security really is a different animal. While most software designers and developers can kick back once their code does what it is supposed to do, we have just started – the important part is what the code also does *not* do. For other applications, added functionality brings cheers from users – more bells and whistles are always better. We add functionality and our users cringe – more restrictions. If we are good, no one will notice we have added more, while our counterparts fly banners with their latest new features. Is it any wonder we get no respect?

In the standards world, we are treated in a similar fashion. We in the POSIX Security Working Group (P13.6) have the unenviable job of policing the work of other POSIX groups, to be sure that gaping security holes are not mandated in the standards. That makes us many friends. We add

interfaces that have sweeping effects on well-established sets of interfaces. We change those pillars of POSIX interfaces and utilities to accommodate our added features. Our job never ends. As new standards are developed we continue to study them for the impact on the security of POSIX-conformant systems. We have just started studying what security means when systems are interconnected. The concepts of user identification and authentication and data markings become remarkably complex once they are taken out of a single system and spread throughout a network. We have a lot of work to do in getting standards that meet the needs of the market and protect the information of those using the end product.

The Great Thing About Standards is There Are So Many To Choose From

Not so many years ago computer standardization was a foreign and even ridiculous thought. In the eighties, however, we started moving toward a more friendly world in which everyone wanted to talk in the same language. Organizations that previously held design and implementation information excruciatingly close soon started sharing their gems freely. Security joined right in: standards were written for what security should be, first in individual countries and then in international cooperation. The utopian view was that someday (soon) there would be a single security standard for the globe. Working groups were formed to look at standardization of security interfaces, utilities, and data. But with limited coordination among groups and with the current problem of downsizing, organizations now send fewer representatives. Each group lacks the resources to make progress on their standards. Substantive progress might be made by pooling resources, and there would be one accepted standard instead of a handful of incomplete ones.

Progress of POSIX Security Working Group

Now I can tell you what we have accomplished. A third ballot of the five initial security options for POSIX.1 (access control lists, mandatory access control labels, information labels, audit and fine-grained privileges) is being distributed as you read this. However, it is about four months behind schedule due to loss of half of the ballot-resolution team. In addition, we have identified several interface areas that we need to tackle in order to complete a set of security interfaces for portable applications (identification and authentication, administration and portable formats of security attributes, cryptography, and network security interfaces). We have been unable to make headway in these new areas because we cannot

get enough organizations to submit proposals, nor can we reach enough people willing to do the work.

What's a chair to do? Flood the Internet with calls for participation and proposals? Done it. Personal appeals to ex-members? Done it. Complain and wallow in self-pity? Done it. Get mad and stomp around some Marriott? Done it. Ignore the problem and act like fifty new attendees will show up? Done it. Continue the work and make progress, no matter how slow? Doing it, for as long as it takes.

Standards Update, What Next?

by Nicholas M. Stoughton

<nick@usenix.org>

Steady-State Behavior

Three years ago, attendance at POSIX meetings was around 340 people. At Irvine this January, the number had fallen to 165, and it is expected (hoped?) that it will bottom out at around 150.

So what has happened? Where have all the contributors gone? There is never a simple answer to questions like these, and there are at least two major influences at work in POSIX attendance. The first is straight economics. The world is in a recession. If your company is losing millions or even billions of dollars each year, is sending people to POSIX meetings the best way of spending what money it does have? Why not work at a distance, through the ballot process?

Another key factor in attendance is the type of work that is required. Three years ago, new standards development was in full swing. Now we are settling down to a steady state. Approved standards for POSIX.1, POSIX.2, POSIX.3, POSIX.4 (aka POSIX.1b), POSIX.5 and POSIX.9 are all there. Maintenance work on these is now a major focus, and that takes a different type of person on the working group. At Irvine, a considerable amount of time has been spent dealing with interpretation requests, chiefly for the most blatantly UNIX standards, POSIX.1 and POSIX.2.

Do we, the UNIX user community, care? Let's give up going to these meetings... after all, someone else is sure to do the right thing, aren't they? It can't be all that hard to maintain a standard! Anyway, we don't want standards rammed down our throats at every opportunity.

Unfortunately, the serpent whose name is Invention is lying there, coiled and ready to strike the moment that someone stops saying, "But where's the existing practice?" I've lost track of how many times Jeff Haemer and I have trotted out

that phrase. In the standards world, practice really does make perfect. If standards are to be rammed down our throats, let them at least be palatable ones.

The rules of interpretation say that the approved standard should be interpreted as loosely as possible. If it is actually wrong, it is up to a later revision to fix the wording, but no one can complain in the interim. This will of course lead to lots of "Weirdnix" implementations: systems that claim POSIX conformance, but are about as far removed from UNIX as you can imagine!

So it is necessary to keep a significant presence within POSIX, attempt to restrain and guide the work occurring. Existing documents need revision to clarify the wording and to help prevent the worst excesses of Weirdnix. New POSIX projects are still being introduced, but at a much-reduced level. Many of these are not necessarily "mainstream UNIX" things either — an ADA interface to time synchronisation, Test Methods for POSIX.1b, the Realtime standard (yes, the number has changed, it used to be known as POSIX.4), and so on. Nevertheless, new direct UNIX projects are not unthinkable, and we must be ready to meet these challenges.

To give you a flavor of what is happening in the interpretations process, Andrew Josey, the Vice Chair of Interpretations has put together a chart—see next page.

In the past months new draft guidelines for PASC interpretations have been circulated and a BOF session met at Irvine to discuss them further.

The guidelines attempt to address the issues of timely response, and the issues of the scope of the interpretations. They are being followed now and we hope to see improvement in the process over the next six months.

What Do You Care About Standards?

I would like to take this opportunity to solicit your opinions. What do you think should appear in this column? I was recently invited to submit a series of articles to a prominent Open System Magazine. After I sent in material, I was told, "These are far too POSIX-centric. What about some other standards?"

There are several other areas that might be useful to report on, both in the *de facto* and *de jure* worlds. But rather than trying to read your minds, I'll solicit your suggestions. What else would you like to hear about?

Working Group	Mailing List	Current Position	Outstanding Requests
P1003.1	intrap1003.1@stdsbbs.ieee.org	61 request pre-Irvine, 31 addressed & in progress	30
P2003.1	intrap1003.1@stdsbbs.ieee.org	17 request, 5 complete	12
P1003.2	intrap1003.2@stdsbbs.ieee.org	30 requests pre-Irvine	0
P1003.5	posix-ada-interps@spectre.mitre.org	9 requests being processed	1
P1224	intrap1224@stdsbbs.ieee.org	6 requests have been answered	0
P1224.1	intrap1224.1@stdsbbs.ieee.org	2 requests have been completed	0
P1224.2	intrap1224.2@stdsbbs.ieee.org	3 requests being processed	0
P1327	intrap1224@stdsbbs.ieee.org	3 requests have been answered	0

Thank You

I would like to take this opportunity to publicly thank Michael Hannah, a regular contributor to this column over the years, for all his work with POSIX.9, and for his wit, and his enthusiasm. Michael has been promoted to a new position within Sandia National Labs, running a 2000-node Intel Paragon system. I know he has some stories that would interest SAGE members,

although he will no longer be in a position to continue his work with POSIX. The first article I edited for this column was from Michael, and the ease with which I was able to work with him persuaded me to take on the job permanently. I am sure you will all join me in wishing him every success in the future.

CitySpace

CitySpace is a virtual city model built collaboratively by kids across the Internet. The project invited kids across the country to share stories, pictures, sounds, and models of each other's neighborhoods through an anonymous FTP site, and to assemble them into a navigable, three-dimensional city model. The project is designed for museums, science centers, and after-school workshops and strives to develop network communications and computer graphics skills for kids from diverse backgrounds. As the city model evolves, video output from the collaborative model is netcast to both CitySpace builders and interested visitors via IP multicast and CU-SeeMe.

The first netwide CitySpace event was held in 1993 at the SIGGRAPH SIGKids Learning lab. In November of the same year, the Computer Museum of Boston hosted the CitySpace project in conjunction with Internet tours, a traveling Internet workshop.

The next CitySpace workshop and live netcast will be held March 7-13th at the Exploratorium in San Francisco, CA. Current areas of project development include CitySpace MOO and remote access to the environment via direct multicast.

The CitySpace SIGKids event was made possible by ACM SIGGRAPH, The Four Oaks Foundation, Sun Microsystems, Silicon Graphics, and The San Diego Supercomputing Center. CitySpace at The Computer Museum was made possible by Silicon Graphics, BBN/Nearnet, Software Systems, and The Electronic Visualization Lab at the University of Chicago.

The Exploratorium event will be made possible by the USENIX Association and Silicon Graphics. CitySpace will travel with the Internet Tours workshop through 1994.

For more information contact <info@cityspace.org> or call 212-753-0573.

The Bookworm

by Peter H. Salus

<peter@uunet.uu.net>

A number of interesting books arrived recently, giving me lots to read flying to and from San Francisco and while there.

The one least within my usual ambit is Bruce Schneier's *Applied Cryptography*. This is a thick, heavy volume that is a genuine overview designed for programmers, not for professional mathematicians. Schneier has made two decades of academic research intelligible to mere mortals. So far as I can tell, this is the only book that presents cryptography in such a fashion. Fascinatingly, the chapter on the DES notes that it has been "a worldwide standard" for over a decade. As we all know (a) that this is true and (b) that the US Government still bars export of DES-containing software, we can but smile wryly. At a time when our government appears hellbent on imposing an encryption standard on the field to which it holds the key, understanding cryptography becomes increasingly important. Schneier has done the community a real service.

Tcl/Tk

Tcl – tool command language – is a software package for developing and using GUI applications. Tk is the Tcl extension toolkit for the X Window System. Both were developed by John Ousterhout, who has spoken about them at several USENIX conferences. At long last his exposition is available. It is elaborate, meticulous, interesting, and valuable. Chapter 3, on Tcl syntax, is one of the best expositions of a syntax I have ever read. The book is made up of five parts: two introductory chapters, Part I: The Tcl Language (Chaps. 3-14), Part II: Writing Scripts for Tk (Chaps. 15-27), Part III: Writing Tcl applications in C (Chaps. 28-37), and Part IV: Tk's C Interfaces (Chaps. 38-45). There is also an appendix on installation.

Another Introduction

Christian and Richter is an OK expansion of Richter's earlier UNIX books. It now covers SVR4 and 4.3BSD, though with an SVR4 prejudice (on p. 4, the reader is informed that Sun "in sensible style" modified SunOS to "make it compatible with System V." Sigh. Readers of this column will know how unhappy I am with the obligatory corruptions of history that are found in UNIX books.

This one is wrong even about the dates of the first commercial publications. There is a good chapter on AWK. This aside, I still prefer Harley Hahn's book (McGraw-Hill).

TCP/Inter/IP/Net

Or something like that.

I was looking around the bookstalls at USENIX. It seems to me that while there were a lot of books on the Internet a year ago, the Vice President (no, not O'Dell, the other one!) has really gotten the publishers going. I have not yet been sent "The Green Guide to Internet use for Gay Whales," but fully expect one. It will be bound in edible dried algae. There has also been a spate of books on TCP/IP. I presume this is due to the unannounced, but equally unlamented demise of OSI.

On the other hand, there are a few bright spots: Rich Stevens' fine volume on the protocols and two excellent books by Carl-Mitchell and Quarterman and Quarterman and Carl-Mitchell (in that order) are among these.

The *Practical Internetworking* book really is. It leaves none of its promises unfulfilled. As might be expected from Quarterman, the history of the ARPANET and of the NCP/TCP protocols is right. So are the dates for UUCP, for USENET, and for BITNET. Carl-Mitchell and Quarterman state that they want this book to be "a practical guide for building TCP/IP networks with the UNIX operating system." Mr. Gore's Interstate 2000 will bring ever more professionals onto the Matrix. They are going to need a book like this to get their machines, LANs, etc., functioning on the data highway. Think of this as the AAA Guide to I-2000 and you won't be far off. If the book has any shortcomings, they lie in the references sections (at the end of each chapter), which seem to shun any work by Doug Comer.

The Internet Connection may be the closest that Quarterman and Carl-Mitchell ever get to a how-to-book. But it isn't how-to in the ways that Kehoe or Dern or Krol are: it is a genuine exposition of how to connect a computer or network to the Internet. As such, it contains step-by-step information on finding/obtaining a connection, registering domains, configuring TCP/IP protocols, etc. There is even some security information. And, of course, archie, WWW, WAIS, and lots of other Internet resources. The endpapers contain four color maps of Internet use and growth. If you get *Matrix News*, you've already seen these (and others); they're terrific and the authors and Gretchen Phillips get my eternal thanks for them. The glossary is useful and the "Further Reading" section graciously includes books like Kehoe, Krol, and

LaQuey as well as *Neuromancer* and *Snow Crash*. I found this section more useful in conjunction with the end-of-chapter "references" than merely the latter in *Practical Internetworking*.

While there are typos here and there in both books, I noticed none that weren't self-correcting.

Brief notes

There is a new (third) edition of Brendan Kehoe's *Zen and the Art of the Internet*. It is nearly double the size of the second edition and now includes a couple of cartoons. Even though it is verging on being too large, it remains the best small book for the true beginner.

There is also a new edition of LaQuey's *Internet Companion* (now: *Internet Companion Plus*). The *Plus* is largely the addition of a diskette containing InterCon's WorldLink "access software" ("for your PC or Macintosh") and installation instructions. As I don't run MS-DOS, I couldn't try it out.

Apologies

After the November/December column, Evelyn Leeper pointed out that my review of the two "dictionaries" was confusing: I liked *The New Hacker's Dictionary* (MIT Press) and found *Jargon* (Peachpit Press) useless. Sorry. Second, at USENIX in San Francisco, Eric Allman told me that I had gotten it all wrong, and that Brian Costales had done most

of the work on the *Sendmail* book (ORA). My sincere apologies Brian, it's a fine and valuable piece of work.

Bruce Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code* in C. John Wiley & Sons, 1994. 618pp. ISBN 0-471-59756-2. \$44.95.

John K. Ousterhout, *Tcl and the Tk Toolkit*. Addison-Wesley, 1994. 450pp. ISBN 0-201-63337-X. \$36.75.

Smoot Carl-Mitchell and John S. Quarterman, *Practical Internetworking with TCP/IP and UNIX*. Addison-Wesley, 1993. 476pp. ISBN 0-201-58629-0. \$43.25.

John S. Quarterman and Smoot Carl-Mitchell, *The Internet Connection*. Addison-Wesley, 1994. 271pp. ISBN 0-201-54237-4. \$32.25.

Kaare Christian and Susan Richter, *The UNIX Operating System*. Third edition. John Wiley & Sons, 1994. 576pp. ISBN 0-471-58684-6. \$22.95.

Brendan Kehoe, *Zen and the Art of the Internet*. Third edition. Prentice Hall, 1994. ISBN 0-13-121492-6. \$23.95.

Tracy LaQuey with Jeanne C. Ryer, *The Internet Companion Plus*. Addison-Wesley, 1993. 198pp. + diskette. ISBN 0-201-62719-1. \$19.95.

A Second Look

Doing Business on the Internet: A Second Look by Frank Hecker

<hecker@access.digex.net>

After reading Mary Cronin's book *Doing Business on the Internet* and Peter Salus's hostile review of it, [in the January/February '94 issue of this newsletter] I'd like to offer a second opinion.

While it's reasonable to use (unspecified) misprints and technical errors as one measure of an author's competence and a book's quality (for example, I cringe whenever I read a book with a "forward"), to focus exclusively on these flaws does the prospective reader a disservice. One might as well dismiss the book because it has an ugly cover and leave it at that.

The more important question is whether Cronin's book has significant value for its intended audience (which is for the most part not the membership of USENIX). I believe that it does.

Cronin's target audience is middle and upper-level non-technical management ("illiterati" as Salus puts it) whose primary job goal is making more money for their companies, and whose primary purpose in reading the book would be to learn about ways in which the Internet can help them do that. *Doing Business on the Internet* provides a generally clear high-level overview of the Internet, USENET, and their associated services, explains in business terms (appropriate to readers of *Fortune* and *Forbes*) how companies can use these services both internally and to communicate with customers, prospects, and suppliers, and provides a number of real-life examples illustrating these points.

These examples are useful beyond just as an aid to understanding; they also can help convince management of the value of the Internet. Corporate managers are generally conservative in outlook and often take their cues from their peers at other companies in their industry. Cronin presents a

wide enough range of examples that most corporate managers will find one that relates to their job function and industry, increasing the likelihood that they'll consider Cronin's advice relevant to them.

Of the misspellings and other errors Salus mentions in passing, some reflect simple sloppiness and do little harm to the main message of the book: for example, misspelling of names, incorrectly capitalizing "Computervision," or attributing to Oracle the slogan "the network is the computer." Others are more serious; for example, Cronin uses the term "public domain software" to encompass all types of freely available software, including copylifted software, thus obscuring the question of whether companies should use such software as a base for software development. She also omits from her discussion of Internet security any mention of the controversy over government restrictions on full use of strong encryption technologies, a subject surely of great interest to companies who conduct business worldwide and wish to protect their Internet email and other data.

My response to these flaws is not that the book is worthless but rather that managers reading the

book will need to supplement it with advice from those inside their companies who are already Internet-literate. It's here that I see the book's relevance to USENIX members, especially those (like me) who are trying to convince their companies' upper management that the Internet and USENET can add value in every part of the business, not just in the research or development groups.

In that sense I believe that Cronin's book can be of greater value to the readers of *login*: than any of the flood of (increasingly repetitive) "how to" Internet books. I'm sure that by now most USENIX members know how to use FTP and can find lists of USENET newsgroups. However I would guess that many of them would like their companies to make more and better use of the Net but are not sure how to overcome a management perception that the Internet and USENET are "toys for techies." If you're one of those people, I recommend that you buy a copy of *Doing Business on the Internet*, read it once to get a feel for what upper management might think important, and then pass it on to someone in management you think might be sympathetic to the cause.

Applied Cryptography

Applied Cryptography: Protocols, Algorithms, and Source Code in C by Bruce Schneier. John Wiley & Sons, 1994, 618pp. ISBN 0-471-59756-2. \$44.95

Reviewed by Donald T. Davis
<don@security.ov.com>

Here at OpenVision's security branch (formerly Geer-Zolot Assoc.), we recently got a copy of Bruce Schneier's new book, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. We immediately ordered two more copies, because our security jocks (me included) didn't want to share it. It is encyclopedic, quite readable, and well-informed, and it more or less picks up where Dorothy Denning's classic *Cryptography and Data Security* (Addison-Wesley, '82) left off a dozen years ago. I've often wished lately that such a reference as Schneier's existed.

Schneier covers those topics in data security that touch most closely on the encryption algorithms themselves. Thus, the book doesn't discuss authorization, audit, firewalls, or the recent for-

mal logics for proving protocols correct. As far as I can tell, it does cover everything about authentication and key-distribution—everything. Of the recent flurry of books and articles on data security that I've seen, including some by my old colleagues from Project Athena, and including a couple of others that are still in press, this one has the clearest and most accurate treatment of kerberos.

The book is structured like a reference, but written like an undergrad text. Thus, you can enter it anywhere and make sense of what you find, even if you don't already know the material well. It does not include exercises or end-of-chapter summaries, but does include a bibliography of 908 references. This makes it a good place to go, before you dive into the literature on a topic like zero-knowledge proofs and protocols. Schneier also includes licensing and sourcing addresses for encryption algorithms. The index, unfortunately, is a bit weak (though it is available from the author on the net: schneier@chinet.com). This book would be a bargain at twice the price.

TCP/IP Illustrated

TCP/IP Illustrated, Volume 1: The Protocols
by W. Richard Stevens. Addison Wesley, 1993.
576 pages. ISBN 0-201-63346-9, \$47.50.

Reviewed by Vern Paxson
<vern@ee.lbl.gov>

Addison-Wesley is promoting Richard Stevens' new book, *TCP/IP Illustrated, Volume 1: The Protocols*, as an indispensable reference for the network practitioner. In a nutshell, they're right.

Just a little experience with the internals of networking is enough to make you realize that: (1) it's complicated; (2) it's hard to diagnose what's really going on out there on the wire; and (3) Gee, the RFC standard says that *this* is supposed to happen here, but I'm seeing *that*. Stevens' book addresses all of these concerns. He tackles the inherent complexity of networking by first giving an overview "roadmap" and then carefully exploring each layer of complexity with prose that somehow remains fresh and readable regardless of how large or small the particular technical matter at hand is. He deals with the problem of diagnosing what's actually going on in a network by repeatedly analyzing network behavior using freely-available software that you can ftp for yourself today. And in doing so, he addresses many fine points regarding the interpretations (and in some cases blatant violations) of the RFC standards found in the 4.4BSD, SunOS, Solaris, AIX, BSD/386, and SVR4 operating systems.

As one might expect, the heart of the book lies in the title word "illustrated." The workings of each of the many protocols are brought to life, not with abstract diagrams as to how they *should* work, but with annotated traces of real traffic from the author's 9-node network laboratory. These traces are generated using tcpdump, a freely-available packet capture tool (Stevens lists in one of the appendices anonymous ftp sites for the 25 publicly-available programs and documents used in the text). Indeed, every major topic regarding every protocol discussed in the book is illustrated with a "live" example; this is TCP/IP networking in the Real World.

The book is a thorough study of the entire TCP/IP protocol stack, from the link layer (Ethernet, SLIP and PPP) to the IP layer (routing, subnetting, ARP, RARP, ICMP, the TTL mechanism and "trace-route," the Domain Name System, broadcasting, multicasting) to the transport layer (UDP and TCP) to the application layer (Telnet, FTP, email, NFS). The treatment of TCP is exceptionally thorough, more than 100 pages covering the basics, advanced topics, nooks & crannies, and proposed changes to this complex, industrial-strength protocol. Stevens' expertise here is impressive: two of my colleagues who have worked on modifying some of TCP's congestion-avoidance algorithms have been asked by others to write up their changes in an RFC. One of them picked up my copy of Stevens', flipped to the section discussing their modifications, and said, "Well, now the pressure's off to write it up, it's already here!"

As well an expression of technical expertise, the book reads like a labor of love. Stevens has taken great care in all facets of the book, and it shows. The organization is well thought out and clearly presented; the cross-referencing thorough without being overwhelming; the technical level right on target and presented simply, yet with expertise; the 150-entry annotated bibliography a first-rate resource in its own right; the index, which I eyeballed at about 2,000 entries, remarkably complete; the exercises at the end of most chapters meaningful and thought-provoking (!); and the writing carefully honed, such that I found covering the 500+ pages in my copious spare time an easy and enjoyable task.

TCP/IP Illustrated has already become my most-likely-to-have-the-answer reference book, the first resource I turn to with a networking question. The few holes in its coverage – no discussion of the NNTP network news transfer protocol, nor of IP tunneling – are felt all the more because of how well all the other topics are dealt with. The book is, all publisher hype aside, an instant classic, and I, for one, am thrilled that something like this is now available.

Sendmail by Bryan Costales, with Eric Allman and Neil Rickert, O'Reilly & Associates, Inc., 1993. ISBN: 1-56592-056-2.. 792pp.

Reviewed by David J. Fiander
<davidf@golem.waterloo.on.ca>

I hate *sendmail* – the program, that is. This fact makes it very hard to write this review. I have to keep in mind that I am reviewing Bryan's book, and *not* Eric's program. That said, if you must administer a *sendmail*-based site, then this is the book to keep on your desk. Bryan's extensive *sendmail* experience, along with input from the leaders in active *sendmail* development, ensures that this is the most up-to-date and accurate *sendmail* manual available.

Sendmail is quite a brick, as befits the complete, nay, exhaustive work on the subject. The book is divided into three sections: "Tutorial," "Administration," and "Reference." In fact, it might have been better if O'Reilly had printed the book in two volumes: one containing the tutorial and administration, the other just the reference (Eric probably ignores the first 200 pages of the copy he supposedly keeps on his desk). Splitting *Sendmail* into two volumes would also have allowed Bryan to actually include the text of RFCs 821 and 822 instead of just referring to them. And finally, the lie-flat binding that O'Reilly seems to have finally settled on, after years of experimentation, doesn't really lie very flat when you're reading the first or last hundred pages.

But *Sendmail*'s size is warranted. While you may not use all of it at once, you will need it all eventually. This book has a definite lifecycle. The tutorial is a painless introduction to the basics of how *sendmail* works, and how to create and test a configuration for your site. While working through the tutorial, you create a simple `sendmail.cf` file, which is suitable for the smaller workstations around your office. Every chapter in the tutorial ends with suggested "Things to Try," many of which make on-going administration of your site much simpler, if you implement them. Once you implement these simple suggestions, you are ready to move onto the big problems of *sendmail* administration.

Bryan's suggestion, at the start of the "Administration" section, that you throw out the vendor-provided *sendmail* and build the current sources from the net, is somewhat drastic, and hardly realistic for many of the beleaguered system administrators who stand to benefit the most from this book. This suggestion probably explains why, while differences between IDA, V8, and older *sendmails* are marked, they are not as well-described as I would have liked in this section of the manual.

This minor nit aside, "Administration" will probably be the most-thumbed section of the whole book, for at least a couple of weeks. During this phase of the book's lifecycle, you will spend more time installing software and writing shell scripts than actually worrying about configuring *sendmail*. Once these chapters are out of the way, and *Sendmail* is just ticking along without a problem, you can shelve the book and forget about it until something goes wrong. That's when the last phase of *Sendmail*'s lifecycle starts, and when the "Reference" section and appendices become crucial.

The reference section of *Sendmail* describes everything about how to configure *sendmail*, both in the `sendmail.cf` file, and from the command line. Bryan devotes an entire chapter to each type of `sendmail.cf` directive, and fully three chapters to macros. Because you will probably refer to this part of the book while trying to fight a fire, every chapter in both the "Administration" and "Reference" sections end with "Pitfalls" which describe the common sorts of things that can go wrong. If you see something happening, but can't figure out why, "Pitfalls" will tell you.

I wanted to complain that *Sendmail* doesn't have a "cookbook" section like that in the *Programming Perl* nutshell handbook but, while writing this review, I couldn't find a problem for which it didn't provide an answer. Fortunately, *Sendmail* has a good index. I have to admit that I got my review copy for free, but I am seriously considering spending the \$33 necessary to buy a second copy for home. This book would be worth it for twice the price.

Announcements/Calls for Upcoming Events

On the following pages are Announcements and Calls for Papers (CFPs) for upcoming USENIX events. Watch *comp.org.usenix* for regular postings of updates, student grant applications, registration, and more.

SANS III.....	50
C++ Tutorial Program.....	52
UNIX Applications Development.....	53
Summer '94 Tutorials.....	55
CFP: High-Speed Networking, Berkeley, CA	56
CFP: LISA 1994, San Diego, CA.....	58
CFP: Very High Level Languages (VHLL), Santa Fe, NM.....	60
CFP: Operating Systems Design and Implementation.....	62
CFP: Workshop on Mobile Computing Systems and Applications.....	64

;login: 50 and 100 Years Ago

by Barry Shein

<bzs@world.std.com>

April 1944

This war forced me from my regular cafe on rue St. Morgue in one of the seedier neighborhoods of Paris to a bodega they called El Bizco near Havana. I'd been trying all week to write down a sentence which captured deep-sea fishing as a metaphor for punchcard sorting when a large oaf I did not like and certainly did not know flung himself down at the table next to mine.

"Good evening, Hum" he managed, uninvited, to say in rather poor Spanish, or so I assumed being unable to speak a word of it myself. "Shut up," I remarked, "Unless you can tell me some juicy tid-bit about von Neumann's sordid evening habits or something Turing is currently so obsessed with that it is crumbling his Oxonian veneer and reducing him to the sort of person who might even confide in me. I could use it in this book I am writing. I would claim of course that I heard these stories from them, not you. No one knows or cares about you." This pitiful though poorly washed gentleman began blubbing in a way I have always hated, but I felt a twinge of mercy for him. "No," he elaborated, choking back tears. I couldn't stand it any longer and put him out of his misery with a #5 Abu and Garcia harpoon, damned near ruining its edge. There was nothing left to do but go back to my metaphor after ordering a half carafe of a locally-made sherry, yet another product of questionable breeding.

To Halve or Halve Not,
Reflections on Punchcard Sorting
Andrew "Papa" Humeingway

At the winter conference just held in Little Rock we broadcast Chess Thompson's keynote talk via radio at the local college. It was heard by dozens of students as far as one hundred miles away.

Executive Director's Report

Effie Young

April 1894

At a Works-in-Progress session at the Winter conference in Honesdale, Mr. Nikolai Tesla demonstrated a method for transmitting telegrams without wires. He brought an elaborate prototype device for this demonstration and a rather large diesel generator for electric power. The demonstration went rather well except that upon sending one lengthy telegram, several attendees' hair caught fire. Mr. Tesla was later heard in the hallway during break remarking that he had a new idea for hair-drying and that he was anxious to get back to his laboratory to work on it.

Conference Report

Sir Martin Collinson

A motion was passed by the board of directors allocating monies for a member survey this spring. The board would like to investigate what would make the conferences more attractive to the membership. Some areas of interest to be surveyed include establishing the desirability of livery services, setting up a telephone room at conferences to allow members to call home, and providing dramatic readings from the classics at the evening reception.

Directors' Meeting Minutes

SANS III Program

The Third Annual System Administration, Networking, and Security Conference. Sponsored by the Open Systems Conference Board with the cooperation of SAGE

**April 4-8, 1994 area
Washington, D.C.**

Wednesday: Opening Keynote Presentation

"Computer Crime: Are You At Risk?"
*Scott Charney, Chief, Computer Crime Unit,
Criminal Division, U.S. Department Of Justice*

Thursday

Track A: Security I Plus Email

A Network Perimeter with Secure External Access
*Frederick M. Avolio, Marcus J. Ranum, Trusted
Information Systems*

Who's Trusting Whom? How to Audit and Manage Users' .rhosts Files
Michele Crabb, Nasa Ames Research Center

Campus Email for Everyone: Making It Work in Real Life
Stephen Campbell, Dartmouth College

Track B: Management I

The Manager's Perspective: Looking at System Administrators through the Site Manager's Eyes and Managing the Manager (Invited)
Bill Howell, University Of North Carolina

Track C: Commercial Tools Part I

"Legent's Strategy For Distributed Systems Management"
Andrew Mccasker, Legent Corporation

"Managing the Service Perspective: Where Legacy and Open Systems Meet"
Michael Matthews, Sterling Software

"Complete Control of all Operations Automation Functions across a Heterogeneous Network, from a Single UNIX Workstation"
Rex Thompkins, 4th Dimension Software

Track A: Security II

How to Identify the Most Common Security Holes (Invited)
Matt Bishop, University of California, Davis

Breaking into Banks: Security Lessons Learned From Financial Services (Invited)
Dan Geer, Openvision

Track B: Internet Resources

Internet Information Resources for the System Administrator
Thomas Barrett, Pacific Bell

Internet Discovery and Retrieval Tools (Invited)
Amy K. Kreiling, University Of North Carolina

Track C: Commercial Tools

Monitoring All Your Systems from a Single Point and Extending Unix Security To Really Protect Your Critical Files
Carla Fitzgerald, Computer Associates

Managing the Databases, Computers and Networks in Client/Server Data Centers
Troy Donley, Ecosystems Division Of Compuware

Backup And Recovery, Data Management Services. *Ranga Rangachan, Legato*

Track A: Management

The Dark Side of Distributed Systems Management: When Bad Things Happen to Good Companies (Invited)
E. Scott Menter, President, Enterprise Systems Management, Inc.

The Mainframe is Dead? Migrating to Open Systems (Invited) *Alan Paller, Computer Associates*

Track B: Perl

Applications of Perl (Invited). *Tom Christiansen*

Track C: Network Management

The Future of Network Management
Lance Travis, Open Software Foundation; Representatives from IBM, Hewlett Packard, and Sun Microsystems

Track A: Backup & Configuration Management

Why Did My Backup Fail? Wolfgang Friedrich, Network & System Management Division, H-P

A Simple and Free System for Automated Network Backups
Karl A. Anderson, Nasa Goddard Space Flight Center- and Brian H. Kirouac, Hughes Stx Corp.

Building an Integrated and Enterprise-specific Configuration Management Solution
Mr. Jan Gottschick, Fraunhofer Institute for Software Engineering and Systems Engineering.

Track B: Potpourri

"Make" as a System Administration Tool
Bjorn Satdeva, /sys/admin, Inc.

There are Several Problems in Connection with System Administration Ethics: A Large Group Discussion: *Rob Kolstad, BSDI.*

Track C: Commercial Tools

"Automated Storage Management"
Michael Athas, Netstor

"Storagecenter Storage Management Software For Unix"
Phil Jamieson, Software Partners/32

Hands-on Demonstrations and Reception

Friday

Track A: Heterogeneous Systems And Security Iii

System Management Support in a Multivendor Environment: The Commonwealth Of Independent Systems Model
Randy C. Marchany, Va Tech Computing Center

Experiences with Tripwire: Using Integrity Checkers For Intrusion Detection
Gene H. Kim And Eugene H. Spafford, Coast Laboratory, Purdue University

Track B: Management Ii

How to Find the Right System Adminin 10 Easy Questions (Invited)
Michele Crabb, Nasa

Panel on Managing Mission Critical Commercial Systems
Scott Menter, Enterprise Systems Management, Chairman

Track C: Commercial Tools

Networks and Security
Dan Geer, Openvision

Friday:

Track A: Distributed Vs. Centralized Management

Installing And Managing Remote Sites
Scott Cohan and Steve Miano, Enterprise Systems Management Corporation

The Operator Shell: A Meansof Privilege Distribution under Unix
Michael Neuman And Gary Christoph, Los Alamos National Labs.

Track B: Networking And Security

A New Network for the Cost of One Scsi Cable: A Simple Caching Strategy For Third-party Applications
Hal Pomeranz, Qms Inc.

Guarding the Fortress: Efficient Methods to Monitor Security on 300+ Systems
Michele Crabb, Nasa Ames Research Center

Friday:

The Change Agent Mind Set for Technology Infusion
Kris K. Bennett, Motorola Cellular Infrastructure Group

Win-win Interaction with Users: A Panel
Chairman: *Rob Kolstad, BSDI*

Closing Session: *Rob Kolstad, BSDI*

Registration Information

Please contact the SANS conference office at 719-599-4303 or email to <sans@fedunix.org>.

C++ Program

USENIX Sixth C++ Technical Conference
April 11 - 14, 1994, Cambridge, MA
Advanced Topics Workshop, April 15, 1994

Tutorial Program: April 11-12

Designing and Implementing Effective Classes
Scott Meyers, Software Development Consultant

Object-Oriented Network Programming with C++
Douglas C. Schmidt, University of California, Irvine

Design Patterns – Elements of Reusable Object-Oriented Software
Richard Helm, DMR Group and John Vlissides, IBM T.J. Watson Research Center

Templates, Containers, and Iterators
Andrew Koenig and Rob Murray, AT&T Bell Labs

A C++ Programmer's View of CORBA
Steve Vinoski, Hewlett-Packard

A Taste of Fresco
Mark Linton, Silicon Graphics

Technical Program: April 13-14

Wednesday, April 13

Welcome Chair: *Doug Lea, SUNY Oswego*
Keynote Address: C++: A Better C - For Whom?
L. Peter Deutsch, Artifex Software

EXTENSIBILITY Chair: *Jim Waldo, Sun Microsystems*

The Object Binary Interface: C++ Objects for Evolvable Shared Class Libraries. *Theodore Goldstein, Sun Microsystems; Alan Sloane, SunPro*

A Framework for Building Extensible C++ Class Libraries
Arindam Banerji, Dinesh Kulkarni, David Cohn, University of Notre Dame

Implementing Signatures for C++
Gerald Baumgartner & Vince Russo, Purdue University

COMPILATION Chair: *Michael Tiemann, Cygnus Support*

Base Class Composition with Multiple Derivation and Virtual Bases
Lee Nackman and John Barton, IBM T.J. Watson Labs

Faster Parsing via Prefix Analysis
Martin Carroll, AT&T Bell Laboratories

Static Type Determination for C++
Hemant Pande & Barbara Ryder, Rutgers University

DEBUGGING Chair: *Judy Grass, CNRI*

Supporting Truly Object-Oriented Debugging of C++ Programs. *James Coplien, AT&T Bell Laboratories*

HotWire — A Visual Debugger for C++
Chris Laffra & Ashok Malhotra, IBM T.J. Watson Labs

Thursday, April 14

MEMORY MANAGEMENT Chair: *Erich Gamma, Taligent*

A Customizable Memory Management Framework
Giuseppe Attardi, ICSI; Tito Flagella, Universita di Pisa

Safe, Efficient Garbage Collection for C++
John R. Ellis, Xerox PARC; David L. Detlefs, DEC SR

Panel – Experiences with C++ in Large Systems
Moderator: *Scott Meyers* Panelists: *Wendell Baker, Berkeley CAD Group Kevin Brown, Morgan Stanley David Goldsmith, Taligent*

DESIGN Chair: *Desmond D'Souza, Icon*

Template Base Delegation
Ted Law, IBM Software Solutions

C++ Design and Implementation Challenges in Technology Computer Aided Design Frameworks. *Goodwin Chin, IBM T. J. Watson Labs; Dharini Sitaraman, Chung Yang and Martin Giles, University of Michigan*

An Object-Oriented Framework for Developing Distributed Applications
Douglas Schmidt, University of California, Irvine

TOOLS Chair: *Steve Vinoski, Hewlett Packard*

Interface Translation and Implementation Filtering
Mark Linton, Silicon Graphics; Douglas Pan, Stanford

A Poor Man's Approach to String-Based Interfacing of C++ Objects
Thomas Kofler, Walter Bischofberger; Bruno Schaeffer, Union Bank of Switzerland - UBILAB

Sharing Between Translation Units in C++ Program Databases
Samuel Kendall, Sun Microsystems Labs; Glenn Allinn, CenterLine Software

A Dossier Driven Persistent Objects Facility
Robert Mecklenburg, Charles Clark, Gary Lindstrom and Benny Yih, University of Utah

Advanced Topics Workshop - Friday, April 15, 1994
Organizers: *Jim Waldo, Sun Microsystems Labs; Steve Vinoski, Hewlett Packard*

Registration Information

Please contact the USENIX conference office at 714- 588-8649 or email to <conference@usenix.org>.

UNIX Applications Development

USENIX UNIX Applications Development Symposium

April 25-28, 1994, Toronto, Ontario, Canada

Sponsored by the USENIX Association and in cooperation with UniForum Canada

Program Co-Chairs:

Jim Duncan, *Pennsylvania State University*
Greg Woods, *PlanIX, Inc.*

Program Committee:

Frank Byrum, *Digital Equipment Corporation*
Neil Groundwater, *SunSoft*
Rob Kolstad, *Berkeley Software Design, Inc.*
Evan Leibovitch, *Sound Software*
Peter Renzland, *Ontario Government*
Greg Rose, *RoSecure Software*
Dan Tomlinson, *Compusoft*
Elizabeth Zwicky, *SRI International, Inc.*

Tutorial Program

Monday and Tuesday, April 25 - 26

Monday:

Essential UNIX Programming
Instructor: *Richard Stevens, Consultant*

Intended Audience: Programmers and system administrators who want to learn more about the essentials of UNIX programming. Some programming experience in C is assumed.

This tutorial covers current UNIX programming concepts required for systems programming. It does not cover the basic functions that most programmers are familiar with (open, lseek, standard I/O, etc.). Rather, it focuses on the poorly documented features that tend to be least understood. Although current standards such as POSIX are mentioned, the tutorial focuses on two real-world implementations of the various standards: 4.4BSD and System V Release 4.

The topics covered are:

- Current UNIX standards
- Process control – Race conditions – Sessions – Job control
- Signals – POSIX.1 signal handling – Unreliable signals calls – Interrupted system calls
- Record locking

- I/O multiplexing (select and poll)
- Memory mapped I/O
- Interprocess communication – Comparison of various methods – Stream pipes – Passing file descriptors – Pseudo terminals
- Threads

Richard Stevens is author of the books *TCP/IP Illustrated* (1994), *Advanced Programming in the UNIX Environment* (1992) and *UNIX Network Programming* (1990). He received his PhD in the area of image processing from the University of Arizona in 1982. Currently he is an author and independent consultant residing in Tucson, Arizona.

Tuesday:

UNIX Power Tools o Getting the Most out of UNIX
Instructor: *Rob Kolstad, Berkeley Software Design, Inc.*

Intended Audience: Programmers, managers, and system administrators wanting to learn more about the powerful development tools available on UNIX.

This tutorial covers some of the powerful tools available in UNIX. Knowing how and when to use them enables you to leverage your UNIX system to maximize your development efforts. This can help you efficiently bring applications to rapid deployment and products to market. The primary goal of this course is to provide enough information for the attendee to begin using some of the many available electronic assistants.

This course discusses:

- Perl– prototyping and scripting language that often provides a total solution for many problems
- Tcl/Tk – an appetite-whetting set of examples about a very powerful windowing mechanism
- RCS and CVS – source control and management systems for small and large groups of programmers
- Software Distribution – how to choose the correct media for distributing your product (including CDROM)

- Make – an introduction to the common features of this powerful program building utility.
- Patch – how to create patch distributions for maintaining source files in the field
- Portability – some issues to consider when writing portable software

Dr. Rob Kolstad is President of Berkeley Software Design, Inc., where he manages a handful of engineers scattered across the USA. He teaches system administration in a wide variety of venues in addition to editing this newsletter. Rob served six years on the USENIX board of directors and was instrumental in establishing the popular USENIX Large Installation System Administration (LISA) Conferences. He chaired the recent Winter 93 USENIX conference.

Preliminary Technical Sessions **Wednesday, April 27**

Introductory Remarks
Jim Duncan, Pennsylvania State University

Keynote Address:
Quelling Riots, Earthquakes and Godzilla –
Porting SimCity to X11
Robert Adams, Dux Software

Development Tools

Wrapping DCE/OSF Client/Server Applications
Israel Gold and Uri Shani, IBM Israel Science and Technology

Dagger: A Tool to Generate Program Graphs
Yih-Farn (Robin) Chen, AT&T Bell Laboratories

Application Development With a UIMS
Daniel Klein, Lonewolf Systems

Improving The Environment

SPP - Low Tech, Practical, UNIX Software Portability
John Sellens, University of Waterloo

Creating a Configurable Compiler Driver for System V Release 4
John F. Dooley and Vince Guarna, Motorola Computer Group

Jam – Make(1) Redux
Christopher Seiwald, Ingres Corporation

Invited Talk: Church's Lemma and Funkenhauser's Rebuttal; Designing a Model of the Software Process
David Tilbrook, Mortice Kern Systems

Thursday, April 28

Solving Unique Problems

Experiences with Tripwire: The Evaluation and Writing of a Security Tool
Gene H. Kim, Eugene H. Spafford, Purdue University

Design, Distribution, and Management of Object Oriented Software
Arindam Banerji, David Cohn, and Dinesh Kulkarni, University of Notre Dame

New Approaches For X

Better Widget Design: A Practioner's Approach
Ed Lycklama, KL Group

Implementing a Generalized Drag-and-Drop in X
Cui-Qing Yang and Shrinand Desai, University of North Texas

The Xt Intrinsics as a General Purpose Application Development Platform or A User Interface Toolkit With Optional Users
Jordan M. Hayes, Heuristic Research; Charles A. Ocheret, Investment Management Services

Experiences

Spanning the Technology Generation Gap: Experiences Upgrading a Network Management Application to UNIX
Jay S. Lark, Teknekron Communications Systems; Pierre Osborne, Stentor Resource Centre

Porting and Maintaining with X and Motif; a Project Retrospective
Paul Davey, IXI Ltd.

Software Design for Installability
Steve Simmons, Inland Sea

Panel Session: "Are we doing the right thing?"
Moderator: Rob Kolstad, Berkeley Software Design, Inc.

Closing Remarks
Jim Duncan, Pennsylvania State University

Registration Information

For further symposium information, please contact the USENIX conference office at 714/588-8649 or via email<conference@usenix.org>.

Summer '94 Tutorials

USENIX Summer 1994 Technical Conference June 6-10, 1994 Boston, Massachusetts

Tutorial Program

Choose from among these twenty-one, full-day tutorials covering topics essential to your professional development. These in-depth tutorials deliver authoritative information of immediate usefulness to you. Tutorials are presented by skilled instructors who are hands-on experts in their topic areas. All tutorials offer printed materials, provided at no extra cost, to support your understanding and provide reference at a later time.

Continuing Education Units

The USENIX Association is now a provider of Continuing Education Units (CEUs). Established by the International Association for Continuing Education and Training, the CEU is a nationally recognized standard unit of measure for continuing education and training, used by thousands of organizations across the United States. Completion of any full-day USENIX tutorial qualifies you for 0.6 CEUs. USENIX offers CEUs for its tutorials for a small additional administrative fee. We provide a certificate for each attendee taking a tutorial for CEU credit and maintain transcripts for all CEU students. You may request CEU credit by checking the appropriate box when registering for the conference.

Monday, June 6

9:00 am – 5:00 pm (includes box lunch)

New: Internet Resource Discovery and Development *Edward Vielmetti, MSN Inc.*

Updated: Advanced Topics in UNIX Security
Matt Bishop, University of California, Davis

Topics in System Administration, Part 1
Trent Hein, XOR Computer Systems, and Evi Nemeth, University of Colorado, Boulder

Sendmail Inside and out
Eric Allman, University of California, Berkeley

Essential UNIX Programming
Richard Stevens, Consultant

The DCE Remote Procedure Call System (RPC)
Richard Mackey, Open Software Foundation

UNIX Power Tools – Getting the Most Out of UNIX
Rob Kolstad, Berkeley Software Design, Inc

New: Hubs, Bridges and Routers – The Tools of Networking
Dr. Vincent C. Jones, Consultant

New: An Overview of Perl 5.0
Tom Christiansen, Consultant

New: Windows NT for UNIX Programmers
Douglas Hamilton, Hamilton Laboratories

Tuesday, June 7

9:00 am – 5:00 pm (includes box lunch)

The Law and the Internet
Daniel Appelman, Heller, Ehrman, White and McAuliffe

Achieving Security in an Internet Environment
Rob Kolstad, Berkeley Software Design, Inc. and Tina Darrow, Lawrence Livermore National Laboratory

Topics in System Administration, Part 2
Trent Hein, XOR Computer Systems and Evi Nemeth, University of Colorado, Boulder

Porting to Solaris 2.x
Marc Staveley, Consultant

The Kerberos Approach to Network Security
Daniel Geer, Jon A. Rochlis, OpenVision Technologies

UNIX Network Programming
Richard Stevens, Consultant

Tcl and Tk: A New Approach to X11 and GUI Programming
John Ousterhout, University of California, Berkeley

New: IP Network Administration
William LeFebvre, Northwestern University

Introduction to Threads, POSIX PThreads, and OSF/DCE Threads
Nawaf Bitar, Silicon Graphics, Inc.

Distributed Object Computing with CORBA
David Chappell, Chappell and Associates

New: Special Programming Techniques – Advanced Data Structures
Keith Bostic, University of California, Berkeley, and Margo Seltzer, Harvard University

Registration Information

For further symposium information, please contact the USENIX conference office at 1 714 588-8649 or via email<conference@usenix.org>.

1994 USENIX SYMPOSIUM ON HIGH-SPEED NETWORKING

AUGUST 1-2, 1994
CLAREMONT
HOTEL & RESORT
OAKLAND, CALIFORNIA

REFEREED PAPER SUBMISSIONS

- ◆ Extended abstracts due:
May 2, 1994
- ◆ Notification to authors:
May 16, 1994
- ◆ Camera-ready final papers due:
June 20, 1994

SYMPOSIUM SCHEDULE

Registration Materials Available:

- ◆ June 1994

Monday, August 1

- ◆ Keynote address, followed by technical sessions
- ◆ Reception
- ◆ Birds-of-a-Feather sessions

Tuesday, August 2

- ◆ Technical sessions

FIELD TRIP

Wednesday, August 3

Join us for visits to two high-speed networking testbeds, XUNET/BLANCA and CalREN, in Berkeley.

ANNOUNCEMENT & CALL FOR PAPERS

High-speed, high-capacity networks promise to change profoundly the way we compute. Fast, wide-area networks pose fresh challenges even for mature operating systems, such as UNIX. How will these innovations shape the design of future operating systems? Can we devise applications that fully (and productively) consume the bandwidth at our disposal?

The goals of this symposium are to encourage the UNIX and high-speed networking communities to commingle, to examine the issues and trends in high-speed networking, and to explore the impact of high-speed networks on systems and applications design.

The single-track symposium offers two days of technical presentations (followed by a field trip on the third day). Formally reviewed papers will be presented and published in the *Symposium Proceedings*. A copy of the *Proceedings* will be distributed to all attendees; additional copies may be purchased from the USENIX Association.

SYMPOSIUM TOPICS

We seek presentations of original work on these (and related) topics:

- ◆ Network architectures
- ◆ Operating system support for high-speed networks
- ◆ Protocols
- ◆ Performance
- ◆ Network management
- ◆ Applications
- ◆ Practical experiences

REFEREED PAPER SUBMISSIONS

If you are interested in presenting your work at the symposium, please submit an extended abstract as described below. The extended abstract should represent the final paper in "short form." Its object is to persuade the Program Committee that you will deliver a good 20-25 minute presentation and final paper.

The Committee needs to know that authors:

- ◆ are tackling a significant problem.
- ◆ are familiar with the current literature about the problem.
- ◆ have devised an original solution.
- ◆ have implemented the solution and, if appropriate, have characterized its performance.
- ◆ have drawn appropriate conclusions about what they have learned and why it is important.

Note that the Program Committee considers it unethical to submit the same paper simultaneously to more than one conference or publication, or to submit a paper that has been or will be published elsewhere, without disclosing this information with the submission.

If your paper is accepted, you are expected to provide a full paper in camera-ready form for publication in the *Proceedings* and to present your work at the Symposium.

HOW TO SUBMIT

A typical extended abstract is roughly 2500 words (5 pages). Indicate clearly whether the paper represents a design, an implementation or a system that is in wide use. You are encouraged to include references. Supporting material may be in note or outline form. If you wish, you may supplement the extended abstract with a copy of a full paper.

Please submit one copy of an extended abstract using *at least two* of the following methods:

- ◆ E-mail (*preferred method*) to: net94papers@usenix.org
- ◆ Mail to:
Pat Parseghian, Program Chair
AT&T Bell Laboratories, Room 2C-472
600 Mountain Avenue
PO Box 636
Murray Hill NJ USA 07974-0636
- ◆ FAX to: Pat Parseghian +1 (908) 582-5857

Please, with your submission, include the following information about the author(s):

- ◆ Name (indicate which author will serve as the contact)
- ◆ Affiliation
- ◆ Daytime telephone
- ◆ Postal address
- ◆ E-mail address
- ◆ FAX number

FOR MORE PROGRAM INFORMATION

Refer questions about refereed paper submissions and other program concerns to the Program Chair:

- ◆ Pat Parseghian
Telephone: +1 (908) 582-4229
E-mail: pep@research.att.com

DATES FOR REFEREED PAPER SUBMISSIONS

Extended abstracts due:

May 2, 1994

- ◆ Notification to authors:

May 16, 1994

- ◆ Camera-ready final papers
due: June 20, 1994

PROGRAM COMMITTEE

- ◆ **Program Chair:**

Pat Parseghian,
AT&T Bell Laboratories

Bill Johnston, *Lawrence
Berkeley Laboratory*

Tom Lyon, *Sun Microsystems*

Jeffrey Mogul,
*Digital Equipment Corporation,
Western Research Laboratory*

Gerald Neufeld, *University of
British Columbia*

Lixia Zhang, *Xerox PARC*

FOR REGISTRATION INFORMATION

Materials containing full details of the symposium program, symposium registration fees and forms, and hotel discount and reservation information will be available early June 1994.

If you wish to receive the registration materials, please contact:

- ◆ USENIX Conference Office
22672 Lambert St, Suite 613
Lake Forest, CA USA 92630
Phone: +1 (714) 588-8649
FAX: +1 (714) 588-9706
E-mail:
conference@usenix.org

USENIX

USENIX, the UNIX and Advanced Computing Systems Professional and Technical Association.

8th USENIX SYSTEMS ADMINISTRATION CONFERENCE (LISA VIII)

SEPTEMBER 19-23, 1994

TOWN AND COUNTRY
HOTEL

SAN DIEGO, CALIFORNIA

IMPORTANT DATES

Refereed Paper Submissions:

- ◆ Extended Abstract Submission
Deadline: May 23, 1994
- ◆ Notification to Authors:
June 24, 1994
- ◆ Final Papers Receipt Deadline:
August 1, 1994

Registration Materials Available:

- ◆ July, 1994

USENIX®

The UNIX and Advanced Computing Systems
Professional and Technical Association

SAGE

THE SYSTEM ADMINISTRATORS GUILD

ANNOUNCEMENT & CALL FOR PARTICIPATION

CO-SPONSORED BY SAGE, THE SYSTEM ADMINISTRATORS GUILD

The annual USENIX Systems Administration Conference provides a forum in which system administrators meet to share ideas and experiences. A growing success for the previous seven years, the USENIX Systems Administration Conference is the only conference which focuses specifically on the needs of system administrators. Its scope includes system administrators from sites of all sizes and configurations.

"Automation: Managing the Computer of the 90's" is the theme of this year's conference. The conference will focus on tools to help system administrators automate administration tasks and troubleshoot problems.

TUTORIAL PROGRAM

◆ Monday and Tuesday, September 19-20, 1994

The two-day tutorial program at the conference offers multiple tracks, with a total of as many as twelve half-day tutorials. Attendees may move between tracks, choosing the sections of most interest to them. Tutorials offer expert instruction in areas of interest to system administrators, novice through experienced. Topics are expected to include Networking, Advanced System Administration Tools, Solaris & BSD Administration, Perl Programming, System Security, and more.

TECHNICAL SESSIONS

◆ Wednesday through Friday, September 21-23, 1994

The three days of technical sessions program will include refereed paper presentations, invited talks, panels, Works-In-Progress (WIP) reports, and Birds-Of-a-Feather (BOF) sessions. The first track is dedicated to presentations of referred technical papers. Although papers of a traditional technical content are very welcome, the Program Committee is especially seeking papers on areas such as useful tools or solutions to system administration problems. Papers which are tutorial in nature would also be appropriate. The second track of the Technical Sessions will offer invited talks, panels, mini-workshops, and similar presentations, and we seek proposals for these presentation formats as well.

Conference Proceedings, containing all refereed papers and materials from invited talks and workshops, will be distributed to conference attendees. The *Conference Proceedings* will also be available from the USENIX Association following the conference.

VENDOR DISPLAY

◆ Wednesday, September 21, 1994, 3:00 pm-9:00 pm

Well informed vendor representatives will demonstrate products and services useful to systems and network administration at the informal table-top display accompanying the USENIX Systems Administration Conference. If your company would like to participate, please contact Peter Mui at (510) 528-8649; FAX (510) 548-8649; E-mail: display@usenix.org

CONFERENCE TOPICS

The Program Committee invites you to submit to the refereed paper track of the technical sessions, as well as submit informal proposals, ideas, or suggestions for the various presentation formats of the second track, on any of the following or related topics:

- ◆ Automating Administration Tasks
- ◆ Distributed System Administration

- ◆ Problem Tracking
- ◆ Predicting problems before they happen
- ◆ System Administration standards
- ◆ Differences in OSF, Solaris, and ?
- ◆ Case studies – “This is the problem we solved and how we solved it.”
- ◆ Career paths for system administrators (“Is there life after support?”)
- ◆ Applications using emerging technology (C++, AI, etc.)
- ◆ Performance Monitoring
- ◆ Hardware-related topics: all about memory, installing disk drives
- ◆ Tools – Useful programs or solutions you have developed and wish to share

REFEREED PAPER SUBMISSIONS

We strongly urge you to request a sample extended abstract by sending e-mail to sample-abstract@usenix.org or telephoning +1 (510) 528-8649.

The Program Committee requires that an extended abstract be submitted for the paper selection process. (Full-papers are not acceptable for this stage; if you send a full paper, you must also include an extended abstract for evaluation.) Your extended abstract should consist of a traditional abstract which summarizes the content/ideas of the entire paper, followed by a skeletal outline of the full paper.

Submissions will be judged on the following criteria: relevancy of topic, quality of work, and quality of the written submission.

Note that the USENIX conference, like most conferences and journals, considers it unethical to submit the same paper simultaneously to more than one conference or publication or to submit a paper that has been or will be published elsewhere.

Authors of an accepted paper will present their paper at the conference and provide a final paper for publication in the *Conference Proceedings*. Final papers are limited to 20 pages, including diagrams, figures and appendix and must be in troff or ASCII format. We will supply you with instructions and troff macros. Papers should include a brief description of the site (if applicable).

WHERE TO SEND SUBMISSIONS

For submission to the refereed paper track, please send submissions by *at least two* of the following methods:

- ◆ (Preferred method) electronic (nroff/troff or ASCII) submission of the extended abstract; e-mail to: dinah@usenix.org
- ◆ FAX to the USENIX Association +1 (510) 548-5738
- ◆ Mail to: LISA 8 Conference, USENIX Association, 2560 Ninth St., Suite 215, Berkeley, CA USA 94710

For submission of all proposals other than extended abstracts of refereed papers, and for inquiries regarding the content of the conference program, contact the Program Chair: Dinah McNutt, Tivoli Systems, P.O. Box 202253, Austin, TX USA 78720-2253, +1 (512) 267-9381, E-mail: dinah@usenix.org

DATES FOR REFEREED PAPER SUBMISSIONS

- ◆ Extended Abstract Submission
Deadline: May 23, 1994
- ◆ Notification to Authors:
June 24, 1994
- ◆ Final Papers Receipt Deadline:
August 1, 1994

PROGRAM COMMITTEE

- ◆ **Program Chair:**
Dinah McNutt,
Zilker Internet Park, Inc.
- Tom Christiansen, *Consultant*
- Trent Hein, *XOR Network Engineering*
- William (Bill) LeFebvre,
Northwestern University
- Pat Parseghian, *AT&T Bell Laboratories*
- Hal Stern, *Sun Microsystems*
- Jeff Tate, *Bank of America*
- Mark Verber, *Xerox PARC*
- Neil Todd, *Swiss Bank Corporation*

FOR REGISTRATION INFORMATION

Materials containing all details of the symposium program, symposium registration fees and forms, and hotel discount and reservation information will be mailed and posted to the net beginning July 1994. If you wish to receive registration materials, please contact:

- ◆ USENIX Conference Office
22672 Lambert Street, Suite 613
Lake Forest, CA USA 92630
+1 (714) 588-8649
FAX: +1 (714) 588-9706
E-mail: conference@usenix.org

VHLL USENIX SYMPOSIUM ON VERY HIGH LEVEL LANGUAGES

OCTOBER 26-28, 1994
EL DORADO HOTEL
SANTA FE, NEW MEXICO

IMPORTANT DATES

DATES FOR REFEREED PAPER SUBMISSIONS:

- ◆ Extended Abstracts Due:
June 30, 1994
- ◆ Notifications to Authors:
July 27, 1994
- ◆ Final Papers Due: Sept. 12, 1994

REGISTRATION MATERIALS AVAILABLE:

- ◆ August, 1994



ANNOUNCEMENT & CALL FOR PARTICIPATION

Using very high level languages (VHLLs), programmers can assemble entire applications from large building blocks in just a small fraction of the time required if conventional programming strategies were used. These languages allow programmers to take advantage of increasingly available hardware cycles, trading cheap machine time for costly programmer time. Thus, VHLLs offer one of the most promising approaches toward radically improving programmer productivity.

UNIX has long supported very high level languages; consider *awk* and the various shells. Often programmers create what are essentially new little languages whenever a problem appears of sufficient complexity to merit a higher level programming interface – consider *sendmail.cf*. In recent years many UNIX programmers have been turning to VHLLs for both rapid prototypes and complete applications. They take advantage of these languages' higher level of abstraction to complete projects more rapidly and more easily than they could have using lower-level languages.

Some VHLLs such as *TCL*, *Perl*, *Icon*, and *REXX* have gained widespread use and popularity. Many others never see the public light. Some of these languages are special purpose, addressing a limited-problem domain (such as graphics, text processing, or mathematical modeling) using powerful primitives created for that specific problem. Other VHLLs are more general purpose in nature, but still much higher level than most traditional compiled languages. Some are stand-alone languages, while others are designed to be embedded in other programs. Many are interpreted, although some are compiled to native machine code; a few occupy a gap between both worlds.

SYMPOSIUM SCOPE AND FORMAT

The USENIX Symposium on Very High Level Languages will spotlight these languages and their usefulness in leveraging certain kinds of tasks. The Symposium will introduce participants to concepts and approaches they haven't examined yet, and publish original work in these areas. Programmers will learn about the relative strengths and weaknesses and extract the key concepts that run through the various languages presented.

The USENIX Symposium on Very High Level Languages will run three days:

- ◆ Wednesday, October 26, will feature hour-long overviews by invited speakers of some of the more popular VHLLs in use today, such as *TCL*, *Perl*, *Icon*, and *REXX*.
- ◆ Thursday and Friday, October 27-28, will consist of refereed papers, tutorial-style invited talks on related topics, and panel discussions.
- ◆ Birds-of-a-Feather sessions will be held Wednesday and Thursday evenings, and a Reception will be held Thursday evening.

Papers on brand-new languages, on existing languages about which little or nothing has been published, on applications that use these languages in creative fashions not yet seen, and on experiences at extending existing languages (for example, adding windowing capabilities to *awk*) are all welcome. Papers should address designing, building, testing, debugging, and measuring the performance and usability of these languages, as well as reference and compare related work in the area. Mention both advantages and disadvantages of the approach selected. For applications using these languages, compare and contrast the design, development, and support effort that were required with this approach versus one using a lower-level language. Good papers will be of interest to people who use other VHLLs than the one described in the paper. For example, a paper describing a system built in a particular language will be much more interesting if it highlights some important feature of the language or problems with the language, or some issue relevant to VHLLs in general.

(continued on reverse side)

PROGRAM COMMITTEE

◆ PROGRAM CHAIR:

Tom Christiansen, *Consultant*
Stephen C. Johnson,
Melismatic Software
Brian Kernighan,
AT&T Bell Laboratories
John Ousterhout,
University of California, Berkeley
Henry Spencer,
University of Toronto

D A T E S F O R R E F E E R E D P A P E R S U B M I S S I O N S

- ◆ Extended Abstracts Due:
June 30, 1994
- ◆ Notifications to Authors:
July 27, 1994
- ◆ Final Papers Due:
September 12, 1994



HOW TO SUBMIT TO THE SYMPOSIUM

Persons interested in participating in panel discussions or organizing Birds-of-a-Feather sessions should contact the program chair as indicated below.

Submissions of papers to be presented at the Symposium and published in the *Symposium Proceedings* must be in the form of an extended abstract. The extended abstract should be 1500–2500 words (3–5 pages) and must be received by June 30, 1994. (If you do send a full paper, you must also include an extended abstract for evaluation.) The extended abstract should represent your paper in short form. Its purpose is to convince the program committee that a good paper and presentation will result. You should show that you are addressing an interesting problem, have surveyed existing solutions, have devised an innovative, and original solution, and have drawn appropriate conclusions about what has been learned.

All submissions should indicate the electronic mail address and telephone number of a principal contact. Authors will be notified of acceptance by July 27, 1994, and will be provided with guidelines for preparing camera-ready copy of the final paper. The final paper must be received no later than September 12, 1994. Note that the USENIX conference, like most conferences and journals, considers it unethical to submit the same paper simultaneously to more than one conference or publication or to submit a paper that has been or will be published elsewhere.

Please submit your extended abstracts to the program chair as follows:

EMAILED SUBMISSIONS (PREFERRED):

- ◆ must be in ASCII, troff (with the -me macro set or raw troff preferred), or Postscript form
- ◆ send to tchrist@usenix.org

HARD COPY SUBMISSIONS:

- ◆ via FAX to +1 (303) 442-7177 (Please refer to Tom Christiansen)
- ◆ via postal mail, please submit 6 paper copies to:

Tom Christiansen
USENIX VHLL Symposium
2227 Canyon Blvd., #262
Boulder, CO 80302

FOR PROGRAM AND REGISTRATION INFORMATION

Materials containing full details of the symposium program, registration fees and forms, and hotel discount and reservation information will be mailed and posted to the net in August 1994. If you wish to receive these materials, please contact:

USENIX Conference Office
22672 Lambert Street, Suite 613
Lake Forest, CA USA 92630
+1 (714) 588-8649; FAX: +1 (714) 588-9706
Internet: conference@usenix.org

USENIX, the UNIX and Advanced Computing Systems
Professional and Technical Association.

USENIX SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION (OSDI)

NOVEMBER 14-18, 1994
MONTEREY, CALIFORNIA

PROGRAM COMMITTEE

♦ Jay Lepreau, *University of Utah,*
Chair

Brian Bershad, *University of*
Washington

David Black, *OSF Research Institute*

Paul Leach, *Microsoft Corporation*

Jim Lipkis, *Chorus Systèmes*

Karin Petersen, *Xerox PARC*

Larry Peterson, *University of Arizona*

Karsten Schwan, *Georgia Institute*
of Technology

Michael Scott, *University of*
Rochester

Willy Zwaenepoel, *Rice University*



The UNIX® and Advanced Computing Systems
Professional and Technical Association

CO-SPONSORED (PENDING) BY
ACM SIGOPS
AND
IEEE TCOS

ANNOUNCEMENT / CALL FOR PAPERS

The OSDI symposium will emphasize both *innovative research and quantified experience* in operating systems. We seek papers describing original work concerning the design, implementation, and use of modern operating systems. Besides mature work, we encourage submissions describing exceptionally promising, speculative work or enlightening negative results.

OSDI has evolved from the combination of three series of USENIX symposia on modern operating systems: "Microkernels and Other Kernel Architectures," "Mach," and "Experiences with Distributed and Multiprocessor Systems." Although papers in the ancestral areas are emphatically solicited, we do not favor any particular OS architecture. In general, OSDI has a broader charter than its predecessors, but this first symposium will retain an emphasis on practical issues relating to OS architecture, distributed systems, multiprocessing, and parallel computing.

SYMPOSIUM TOPICS

Questions regarding a topic's suitability are welcome and should be sent via e-mail to the program chair at lepreau@cs.utah.edu

Topics of interest include, but are not limited to:

- ♦ OS structure and organization
- ♦ Microkernel internals, servers, and applications
- ♦ Multiprocessor and MPP issues
- ♦ Scalability and high availability
- ♦ Distributed and mobile systems
- ♦ Object-orientation in OS's
- ♦ Security in distributed systems
- ♦ Communication paradigms
- ♦ Performance and optimization
- ♦ OS support for real time and multimedia
- ♦ OS support for embedded systems
- ♦ OS interaction with HW architecture

SYMPOSIUM OVERVIEW

The OSDI symposium will offer one full day of tutorials, in which a choice of tutorials presented by expert instructors will be available. Following will be two and one half days of technical sessions, with presentations of papers refereed by the Program Committee. The papers will be published in the *Proceedings*, provided free to technical session attendees and available for purchase from USENIX. Selected papers of particular merit, or possibly the *Proceedings*, will likely be distributed to ACM SIGOPS and IEEE TCOS members. A work-in-progress session will be held and will be described in later announcements.

REFEREED PAPER SUBMISSIONS:

- ♦ Extended abstracts due: June 21, 1994
- ♦ Notification to authors: August 5, 1994
- ♦ Full papers due for editorial review: September 6, 1994
- ♦ Camera-ready full papers due: October 4, 1994

REFEREED PAPERS – WHAT TO SUBMIT

Authors must submit an extended abstract by June 21, 1994. The extended abstract should be 5–7 pages long or about 2500–3500 words, not counting references and figures. Longer abstracts will be penalized in the review process. The full papers resulting from accepted abstracts will go through an editorial review cycle with a member of the program committee, and should end up about 10–14 pages long. Very similar papers must not have been published or submitted for publication elsewhere. Papers accompanied by so-called “non-disclosure agreement” forms are not acceptable and will be returned to the author(s) unread. Note that all submissions are held in the highest confidentiality prior to publication in the *Proceedings*.

The object of an extended abstract is to convince the reviewers that a good paper and 25-minute presentation will result. It is important to identify what has been accomplished, to explain why it is significant, and to compare with prior work in the field, demonstrating knowledge of the relevant literature. The extended abstract should represent the paper in “short form.” It must include the abstract as it will appear in the final paper. The body of the extended abstract should be complete paragraphs, not just an outline of the paper. (Sections present in the full paper but omitted from the abstract may be summarized in terse form.) Authors should include full references, figures when available, and, as is usually appropriate, performance data. Such data also help indicate the status of the implementation, often a crucial issue. The abstract will be judged on significance, originality, clarity, relevance, and correctness.

Potential authors are encouraged to contact osdi-info@usenix.org or +1 510-528-8649 to receive additional guidance about the submission and review process.

For administrative reasons (not blind reviewing), every submission should include one additional page containing:

- ♦ paper title and authors, indicating any that are full time students, and
- ♦ for the author who will act as the contact to the program committee, his or her name, paper mail address, daytime and evening phone numbers, e-mail address and fax number, if available.

WHERE TO SUBMIT

Please send one copy of an extended abstract to the program chair via one of the following methods. All submissions will be acknowledged.

- ♦ Paper mail (preferred method) to:
Jay Lepreau, Department of Computer Science
3190 M.E.B., University of Utah
Salt Lake City, UT, USA 84112
- ♦ E-mail (Postscript or ASCII) to: osdi-papers@usenix.org

IMPORTANT DATES

Refereed Paper Submissions:

- ♦ Extended abstracts due:
June 21, 1994
- ♦ Notification to authors:
August 5, 1994
- ♦ Full papers due for editorial review: September 6, 1994
- ♦ Camera-ready full papers due:
October 4, 1994

REGISTRATION MATERIALS

Materials containing all details of the technical and tutorial programs, registration fees and forms, and hotel information will be mailed beginning in August 1994. If you wish to receive the registration materials, please contact:

- ♦ USENIX Conference Office
22672 Lambert Street,
Suite 613
Lake Forest, CA USA 92630
+1 714-588-8649
Fax: +1 714-588-9706
Internet:
conference@usenix.org

Call for Participation

Workshop On Mobile Computing Systems and Applications

December 8-9, 1994

Santa Cruz, CA

Sponsored by the IEEE Computer Society TCOS
(pending) in cooperation with ACM SIGOPS and
USENIX Association

General Chair

Darrell Long, University of California, Santa Cruz

Program Chair

M. Satyanarayanan, Carnegie Mellon University

Exhibits

Peter Honeyman, University of Michigan

Finance & Registration

Richard Golding, Hewlett-Packard

Publication

Luis-Felipe Cabrera, IBM Almaden

Program Committee

Dan Duchamp, Columbia University

Peter Honeyman, University of Michigan

Randy Katz, UC Berkeley and ARPA

Jay Kistler, DEC SRC

Krishan Sabnani, AT&T

*Holmdel M. Satyanarayanan, Carnegie Mellon
University*

Amal Shaheen, IBM Austin

Marvin Theimer, Xerox PARC

Rich Wolff, Bellcore

A major challenge of this decade is the effective exploitation of two symbiotic technologies: portable computers and wireless networks. Harnessing these technologies will dramatically change the computing landscape. But realizing the full potential of the resulting mobile computing systems will require advances in many areas such as:

hardware • communications • scalability •
power management • security • data access
• user interfaces • location sensitivity

The goal of this workshop is to foster exchange of ideas in mobile computing among workers in the field. Attendance will be limited to about 60 participants, based on the position papers submitted. Submissions should be fewer than 5 pages in

length and may expose a new problem, advocate a specific solution, or report on actual experience.

In addition, we will be hosting a small number of novel hardware and software exhibits relevant to mobile computing. The exhibits may be research prototypes or commercial products. Interested parties should submit technical descriptions of their exhibits.

Online copies of the position papers will be made available via anonymous FTP prior to the workshop. A printed proceedings will be published after the workshop, and mailed to participants.

A small number of graduate students will be granted a waiver of the registration fee. In return, these students will be required to take notes at the workshop and help put together the proceedings. Students who wish to be considered for the waiver must send in a brief description of their current research, and an explanation of how participation in the workshop is likely to help them.

Send 10 copies of position papers to:

M. Satyanarayanan
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
Email: satya@cs.cmu.edu
Phone: (412)-268-3743
Fax: (412)-681-5739

Send exhibit descriptions to:

Peter Honeyman
CITI
University of Michigan
Ann Arbor, MI 48103-4943
Email: honey@citi.umich.edu
Phone: (313)-763-4413
Fax: (313)-763-4434

Important Dates

Submissions due August 20, 1994

Acceptance Notification October 1, 1994

Camera-ready copy due November 15, 1994

Publications from John Wiley & Sons, Inc.

Member Number: _____

All USENIX members receive a 15% discount on the following Wiley publications

Introduction to Client/Server Systems

Paul Renaud

1-57774-X \$34.95 *member price: \$29.71*

of Copies: _____

Portable UNIX

Douglas Topham

1-57926-2 \$14.95 *member price: \$12.71*

of Copies: _____

The UNIX Command Reference Guide

Kaare Christian

1-85580-4 \$32.95 *member price: \$28.01*

of Copies: _____

C++ For Programmers

Leendert Ammeraal

1-93011-3 \$32.95 *member price: \$28.01*

of Copies: _____

Obfuscated C and Other Mysteries

Don Libes

1-57805-3 \$39.95 *member price: \$33.96*

of Copies: _____

Adventures in UNIX Network Applications Programming

Bill Rieken

1-52858-7 \$34.95 *member price: \$29.71*

of Copies: _____

UNIX, Self Teaching Guide

George Leach

1-57924-6 \$19.95 *member price: \$16.96*

of Copies: _____

UNIX Shell Programming, Second Edition

Lowell Jay Arhur

1-51821-2 \$29.95 *member price: \$25.46*

of Copies: _____

Object Oriented Programming with Turbo C++

Keith Weiskamp

1-52466-2 \$24.95 *member price: \$21.21*

of Copies: _____

Berkeley UNIX

A Simple & Comprehensive Guide

James Wilson

1-61582-X \$37.95 *member price: \$32.26*

of Copies: _____

U.S. orders only

- ☐ Payment enclosed, plus sales tax
☐ VISA ☐ Mastercard
☐ American Express
Card No. _____

Please mail or fax orders to the following address:

John Wiley & Sons, Inc.
605 Third Avenue
New York, NY 10158
Attn: Karen Cooper

Name _____
Firm _____
Address _____
City/State/Zip _____
Signature _____
(order invalid unless signed)

Phone: (212) 850-6789
FAX: (212) 850-6142





PTR Prentice Hall is pleased to recommend the following titles to USENIX members...

**USENIX members receive a
15% discount on orders**

— **Object-Oriented Programming**, Peter Coad and Jill Nicola,
0-13-032616-X
(03261-5) List: \$47.00 Members: \$39.95

— **Internetworking with TCP/IP, Vol. III Client Server
Programming and Applications for the BSD Socket
Version**, Douglas E. Comer and David L. Stevens,
0-13-474222-2
(47422-1) List: \$51.00 Members: \$43.35

— **Internetworking with TCP/IP, Vol. III Client Server
Programming and Applications for the AT&T TLI
Version**, Douglas E. Comer and David L. Stevens,
0-13-474230-3
(47423-9) List: \$50.00 Members: \$42.50

— **The Internet Message: Closing the Book with Electronic
Mail**, Marshall T. Rose, 0-13-092941-7
(09294-0) List: \$46.00 Members: \$39.10

— **Object-Oriented Databases: An Introduction**,
Dimitris N. Chorafas and Heinrich Steinmann, 0-13-491804-5
(49180-3) List: \$36.00 Members: \$30.60

— **The Anatomy of Programming Languages**
Alice E. Fischer and Frances S. Grodzinsky, 0-13-035155-5
(03515-4) List: \$55.00 Members: \$46.75

— **The Standard C Library**, PJ Plauger, 0-13-131509-9
(13150-8) List: \$35.00 Members: \$29.75

— **All About Administering the NIS+**
Rick Ramsey, 0-13-068800-2
(06880-9) List: \$35.00 Members: \$29.75

— **The Simple Book: An Introduction to Internet Management**
Marshall T. Rose, 0-13-177254-6
(17725-3) List: \$55.00 Members: \$46.75

— **Fiber Optics Networks**,
Paul Green, Jr., 0-13-319492-2
(31949-1) List: \$62.00 Members: \$52.70

— **Networking Operations on UNIX SVR4**,
Mike Padavano, 0-13-613555-2
(61355-4) List: \$44.00 Members: \$37.40

— **The User's Directory of Computer Networks**,
Tracy L. LaQuey, 0-13-950262-9
(95026-1) List: \$36.00 Members: \$30.60

— **MIT Project Athena**, George A. Champine, 0-13-585324-9
(58532-3) List: \$29.00 Members: \$24.65

— **The Matrix**, John S. Quarterman, 0-13-565607-9
(56560-6) List: \$50.00 Members: \$42.50

— **Solaris Porting Guide**, Sunsoft ISV Engineering
0-13-030396-8
(03039-5) List: \$49.00 Members: \$41.65

— **Multiprocessor System Architectures**, Catanzaro
0-13-089137-1
(08913-6) List: \$42.00 Members: \$35.70

— **The HP-UX System Administrator's "How To" Book**
Marty Poniatowski, 0-13-099821-4
(09982-0) List: \$32.00 Members: \$27.20

— **UNIX System V Performance Management**, Edited by
Phyllis Eve Bregman and Sally A. Browning
0-13-016429-1
(01642-8) List: \$29.95 Members: \$25.45

— **SCO® UNIX® Operating System System Administrator's
Guide**, Santa Cruz Operation, 0-13-012568-7
(01256-7) List: \$38.00 Members: \$32.30

MAIL, PHONE OR FAX YOUR ORDER!

Mention this code when calling: D582-91529-LF(2)

___ Check enclosed, made payable to: **PTR Prentice Hall**

___ Please bill my: ___ VISA ___ MasterCard ___ AMEX

Card No. _____

Exp. Date _____ Signature _____

SHIP TO:

Name _____

Company _____

Address _____

City _____ State _____ Zip _____

In USA:
PTR Div.—Prentice Hall Publishing
Box 11073
Des Moines, IA 50381-1073
515-284-6751 FAX 515-284-2607

In Canada:
Prentice Hall Canada
1870 Birchmount Rd.
Scarborough, ON M1P 2J7
416-293-3621 FAX 416-299-2540

In UK, Europe, Middle East & Africa:
Simon & Schuster Int'l
Campus 400, Marylands Ave.
Hemel Hempstead, Herts HP2 7EZ
442-88 2163 FAX 442-88 2265

FOR MORE INFORMATION, OR QUANTITY ORDERS, PLEASE CALL 201-592-2657

NET- WORKS

USENIX members receive a 15% discount
on the following MIT Press publications:

GLOBAL NETWORKS

Computers and International
Communication

edited by Linda M. Harasim

Global Networks takes up the host of issues raised by the new networking technology that now links individuals, groups, and organizations in different countries and on different continents. The twenty-one contributions focus on the implementation, application, and impact of computer-mediated communication in a global context.

340 pp. \$29.95 hardcover HARNH

THE NETWORK NATION

Human Communication via Computer
Revised Edition

Starr Roxanne Hiltz and Murray Turoff

"*The Network Nation*... contained a fascinating vision. ...It is a place where thoughts are exchanged easily and democratically and intellect affords one more personal power than a pleasing appearance does. Minorities and women compete on equal terms with white males, and the elderly and handicapped are released from the confines of their infirmities to skim the electronic terrain as swiftly as anyone else." — Teresa Carpenter, *Village Voice*

580 pp. \$24.95 paperback HILWP

THE EVOLUTION OF C++

Language Design in the Marketplace of
Ideas

edited by Jim Waldo

This collection of articles traces the history of C++, from its infancy in the Santa Fe workshop, to its proliferation today as the most popular object-oriented language for microcomputers. Waldo notes in his postscript that in the process of evolving, the language has lost a clearly articulated, generally accepted design center, with no common agreement about what it should or should not do in the future.

279 pp. \$24.95 paperback WALEP

TECHNOLOGY 2001

The Future of Computing and Communications

edited by Derek Leebaert

Researchers, executives, and strategic planners from inside the companies and laboratories that have shaped today's information age forecast the merging technologies that could well define the computing and communications environment that lies ahead.

392 pp. \$14.95 paperback LEEEP

THE DIGITAL WORD

Text-Based Computing in the Humanities

edited by George P. Landow and Paul Delany

This book explores the larger realm of the knowledge infrastructure where texts are received, reconstructed, and sent over global networks.

Technical Communication and Information Systems series 384 pp. \$39.95 hardcover LANDH

SOCIOMEDIA

Multimedia, Hypermedia, and the Social
Construction of Knowledge

edited by Edward Barrett

Sociomedia continues the assessment of hypertext and hypermedia systems begun in *Text*, *ConText*, and *HyperText* and *The Society of Text*. It examines the use of integrated multimedia to support social or collaborative research, learning, and instruction in the university, one of the best environments for developing and analyzing the effects of computing technologies on our understanding of complex sets of information.

Technical Communications and Information Series 360 pp. \$50.00 hardcover BARRH

CONNECTIONS

New Ways of Working in the Networked
Organization

Lee Sproull and Sara Kiesler

"...Sproull and Kiesler raise crucial questions about our technical and particularly our human strategies as a producing society."

— Howard Webber, *Sloan Management Review*

228 pp. \$21.95 paperback SPRCP

TECHNOBABBLE

John A. Barry

"A serious study of the language of the new technocracy."

— William Safire, *The New York Times Magazine*

288 pp. \$12.50 paperback BARCP

Please send me these titles _____

☐

Payment enclosed

☐

Purchase Order Attached

Charge to my:

☐

Master Card

☐

VISA

Card # _____

exp. _____

Signature _____

\$ _____ Total for book(s)

\$ 3.00 Postage for North American addresses

\$ _____ Canadian customers add 7% GST

\$ _____ Total for book(s) & postage

Make checks payable
and send order to:

THE MIT PRESS

55 Hayward Street, Cambridge, MA
02142-1399 USA

To order by phone, call
(617) 625-8569

or (800) 356-0343. E-mail order

mitpress-orders@mit.edu. The
operator will need this code: **UNIX1.**

Name _____

Address _____

City _____

Phone _____

State _____

Zip _____

FAX _____

McGraw-Hill Books

20% Discount for USENIX Members

	Qty
<i>Open Systems Interconnection Its Architecture and Protocols</i> , Revised Edition.....	<input type="checkbox"/>
Bijendra N. Jain & Ashok K. Agrawala 0-07-032385-2 hardcover retail: \$50.00 member price: \$40.00	
<i>A Students Guide to UNIX</i>	<input type="checkbox"/>
Harley Hahn 0-07-025511-3 paperback retail: \$25.75 member price: \$20.60	
<i>Data Network Design Packet Switching, Frame Relay, 802.6/DQDB</i>	<input type="checkbox"/>
Darren L. Spohn 0-07-060360-X hardcover retail: \$59.00 member price: \$47.20	
<i>Migrating to Open Systems: Taming the Tiger</i>	<input type="checkbox"/>
Daniel R. Perley, 0-07-707778-4 hardcover retail: \$30.00 member price: 24.00	

I am a member of the USENIX Association. Please send me the books I have indicated at the member special rate. I have added \$3.00 postage and handling for the first book ordered, \$1.00 for each additional book, plus my local sales tax.

Total price of books _____
Sales tax _____
Postage & Handling _____
Total enclosed _____

USENIX Membership # _____

____ Check or money order is enclosed - payable to McGraw-Hill, Inc.

____ Charge my ____ Visa ____ Mastercard ____ Discover ____ Amex

Account # _____

Expiration Date _____

Signature _____

Shipping Information (7/93 - 03US001)

Name _____

Address _____

Daytime Phone # _____

Send orders to:

McGraw-Hill, Inc.
Attn: Rosa Perez
11 West 19th Street - 4th Floor
New York, New York 10011

If I am not completely satisfied, I will return the book(s) within 15 days or a full refund or credit.

Satisfaction unconditionally guaranteed. Prices subject to change without notice. We can only accept orders within the continental USA.

THE WHOLE INTERNET



USER'S GUIDE & CATALOG

FROM O'REILLY & ASSOCIATES, INC.

Books that help people get more out of computers



The Whole Internet User's Guide & Catalog

By Ed Krol, 1st Edition September 1992
400 pages, ISBN 1-56592-025-2

A comprehensive—and best-selling—introduction to the Internet, the international network that includes virtually every major computer site in the world. This book is a comprehensive introduction to what's available and how to find it. In addition to electronic mail, file transfer, remote login, and network news, *The Whole Internet* pays special attention to some new tools for helping you find information. Whether you're a researcher, a student, or just someone who likes electronic mail, this book will help you to explore what's possible.

Learning perl

By Randal L. Schwartz
1st Edition November 1993
274 pages, ISBN 1-56592-042-2

Perl is rapidly becoming the "universal scripting language." Combining capabilities of the UNIX shell, the C programming language, *sed*, *awk*, and various other utilities, it has proved its use for tasks ranging from system administration to text processing and distributed computing. *Learning perl* is a step-by-step, hands-on tutorial designed to get you writing useful perl scripts as quickly as possible. In addition to countless code examples, there are numerous programming exercises, with full answers. For a comprehensive and detailed guide to programming with Perl, read O'Reilly's companion book *Programming perl*.

Understanding Japanese Information Processing

By Ken Lunde
1st Edition September 1993
470 pages, ISBN 1-56592-043-0

Understanding Japanese Information Processing provides detailed information on all aspects of handling Japanese text on computer systems. It covers everything from the origins of modern-day Japanese to the latest information on specific emerging computer encoding standards.

10% OFF
TO USENIX MEMBERS
Mention "usenix" & your membership
number when ordering.

sendmail



sendmail

By Bryan Costales, with Eric Allman & Neil Rickert
1st Edition November 1993
830 pages, ISBN 1-56592-056-2

Although *sendmail* is used on almost every UNIX system, it's one of the last great uncharted territories—and most difficult utilities to learn—in UNIX system administration. This book provides a complete *sendmail* tutorial, plus extensive reference material. It covers the BSD, UIUC IDA, and V8 versions of *sendmail*. Part One of the book is a tutorial on understanding *sendmail*; Part Two covers practical issues in *sendmail* administration; Part Three is a comprehensive reference section; and Part Four consists of appendices and a bibliography.

ORACLE Performance Tuning

By Peter Corrigan & Mark Gurry
1st Edition September 1993
642 pages, ISBN 1-56592-048-1

The ORACLE relational database management system is the most popular database system in use today. This book shows you the many things you can do to dramatically increase the performance of your existing ORACLE system, whether you are running RDBMS Version 6 or Version 7. You may find that this book can save you the cost of a new machine; at the very least, it will save you a lot of headaches.

Migrating to Fortran 90

By James F. Kerrigan
1st Edition November 1993 (est.)
315 pages (est.), ISBN 1-56592-049-X

This book is a practical guide to Fortran 90 for the current Fortran programmer. It provides a complete overview of the new features that Fortran 90 has brought to the Fortran standard, with examples and suggestions for use. Topics include array sections, modules, allocatable arrays and pointers, file handling, and numeric precision.

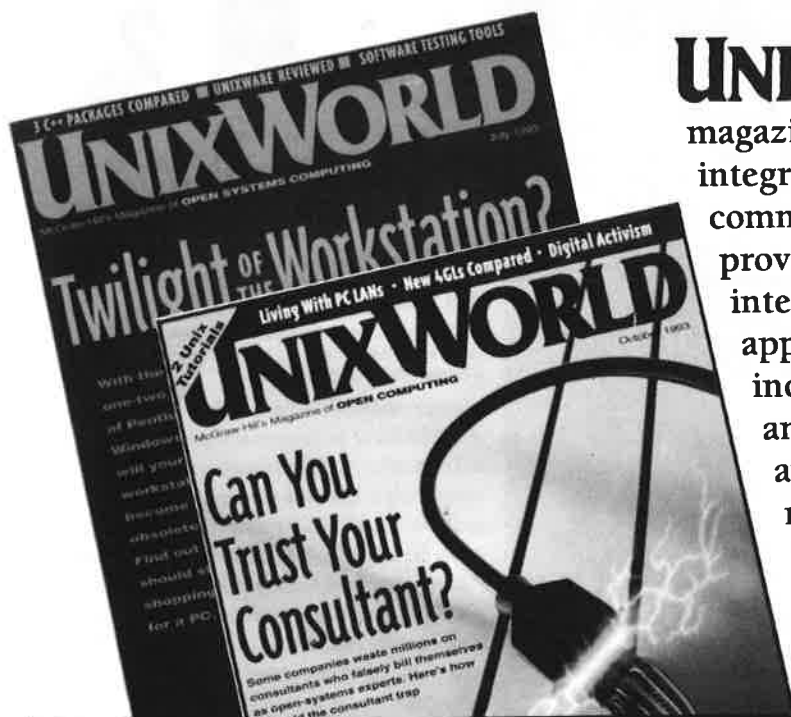
O'Reilly & Associates, Inc.

103A Morris Street, Sebastopol, California 95472 • 800/998-9938 • 707/829-0515 • FAX 707/829-0104 • order@ora.com

Special Offer

THE WORLDWIDE COMPUTING AUTHORITY

USENIX Association Members



UNIXWORLD is a monthly magazine written for people who integrate, resell, manage, and program commercial computer systems to provide solutions based on interoperable networks and applications. Topics for features include the effective use of UNIX and other open systems to downsize and distribute applications. The magazine also provides a useful mix of product reviews, case studies, industry profiles and analysis, and technical tutorials.

THE BENEFITS OF UNIXWORLD:

Programming Tips - helps the reader become educated in UNIX and OPEN SYSTEMS computing. Provides real-world application information the reader can use everyday.

Unbiased Hardware and Software Reviews - helps the reader recommend and justify purchasing decisions.

Clear Editorial - so both beginners and experts can get full benefit from the publication. Product Trends and Technical Advances - to aid professionals in developing company-wide systems strategies.

Profiles on U.S. and International Companies - provides the reader with a truly global perspective of the marketplace and the players. Helps vendors assess and monitor the competition.

Special Rates:

DOMESTIC	CANADIAN	OTHER INTERNATIONAL (Air Delivery)
1 Yr. \$15.00	1 Yr. \$21.00	1 Yr. \$ 48.00
2 Yrs. \$24.00	2 Yrs. \$36.00	2 Yrs. \$ 90.00
3 Yrs. \$30.00	3 Yrs. \$48.00	3 Yrs. \$130.00

Be wise subscribe today!

To place your order please call:

Kelly Kendall at 1-800-327-6850 or fax your order to 609/426-7055



McGraw-Hill, Inc. • Dept. 418 • Hightstown, NJ 08520

Announcing the 1994 Open Systems Products Directory

The value of your UniForum general membership is about to be enhanced with the publication of the 1994 *Open Systems Products Directory*. You'll receive your personal copy with your paid membership (or renewal) to UniForum. Priced to sell to non-members at \$125, the 1994 *Open Systems Products Directory* is a real value. Why spend an extra penny when it's free with paid membership?

For just \$100 member dues you will get *UniForum Monthly* magazine, *UniNews* newsletter, UniForum Technical Publications, Member discounts and the 1994 *Open Systems Products Directory*. Call toll free (800) 255-5620; or (408) 986-8840 outside the U.S. Have your major credit card ready and we'll start your membership at once.

The *Open Systems Products Directory* has been acclaimed by UNIXWorld as "the most comprehensive guide to UNIX products". The 1994 Directory carries almost 7,700 detailed product and service listings — more than ever before. With over 2,100 vendors, the Directory offers 1,600+ pages of: Horizontal and Vertical Software, Communications Systems and Software, Application Development Tools, System Hardware, Operating Systems and Components, Peripherals and Devices, and Services such as Publications, Consultants, Conferences, Training and much, much more.

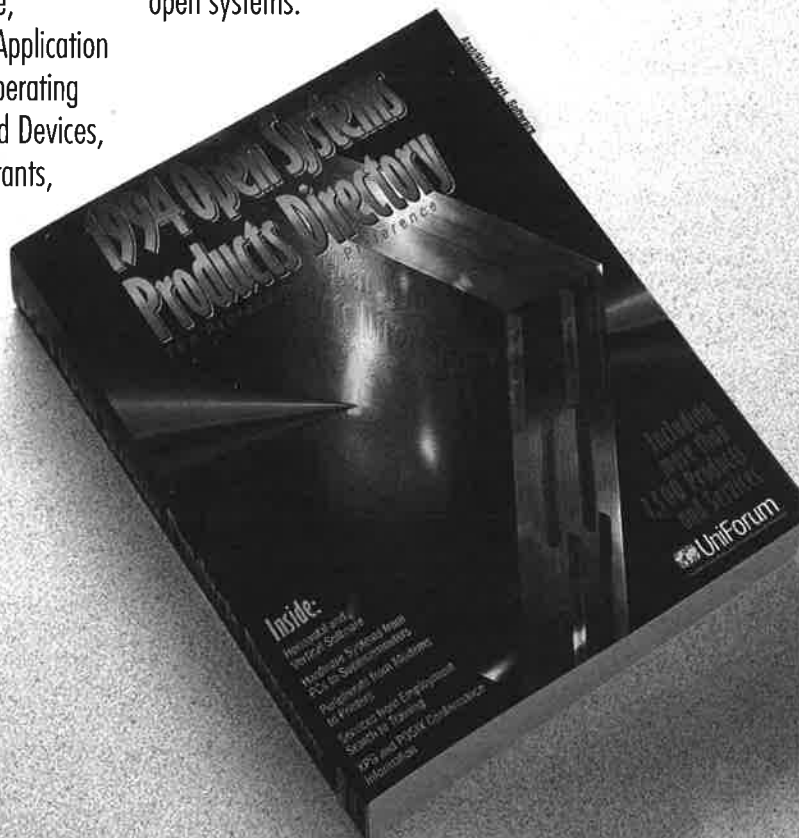
We've added a special Operating Systems Index that makes it easier for you to locate products or services that will run on your existing platforms or those you're planning to use. The Directory is not just a UNIX guide but one that is built comprehensively around open systems.

 UniForum.

2901 Tasman Drive, Suite 205,
Santa Clara, CA 95054

800.255.5620

408.986.8840 Fax: 408.986.1645



Local User Groups

The Association will support local user groups by doing a mailing to assist in the formation of a new group and publishing information on local groups in *;login:*. At least one member of the group must be a current member of the Association. Send additions and corrections to: <login@usenix.org>.

CA – Fresno:

The Central California UNIX Users Group consists of a uucp-based electronic mailing list to which members may post questions or information. For connection information:

Educational and governmental institutions:
Brent Auernheimer (209) 278-2573,
<brent@CSUFresno.edu or csufres!brent>

Commercial institutions or individuals:
Gordon Crumal (209) 251-2648
<csufres!gordon>

CA – Orange County:

Meets the 2nd Monday of each month

UNIX Users Association of Southern California
Paul Muldoon (714) 556-1220 ext. 137
New Horizons Computer Learning Center
1231 E. Dyer Rd., Suite 140
Santa Ana, CA 92705

CO – Boulder:

Meets monthly at different sites. For meeting schedule, send email to <fruug-info@fruug.org>.

Front Range UNIX Users Group
Software Design & Analysis, Inc.
1113 Spruce St., Ste. 500
Boulder, CO 80302
Steve Gaede (303) 444-9100
<gaede@fruug.org>

D.C. – Washington, D.C.:

Meets 1st Tuesday of each month.

Washington Area UNIX Users Group
9811 Mallard Drive
Laurel, MD 20708
Alan Fedder (301) 953-3626

FL – Coral Springs

S. Shaw McQuinn (305) 344-8686
8557 W. Sample Road
Coral Springs, FL 33065

FL – Melbourne:

Meets the 3rd Monday of every month.

Space Coast UNIX User's Group
Steve Lindsey (407) 242-4766
<lindsey@vnet.ibm.com>

FL – Orlando:

Meets the 3rd Thursday of each month.

Central Florida UNIX Users Group
Mikel Manitijs (407) 444-8448
<mikel@aaa.com>

FL – Western:

Meets 1st Thursday of each month.

Florida West Coast UNIX Users Group
Richard Martino (813) 536-1776
Tony Becker (813) 799-1836
<mcrsys!tony>
Ed Gallizzi, Ph.D. (813) 864-8272
<e.gallizzi@compmail.com>
Jay Ts (813) 979-9169
<uunet!pdn!tscs!metran!jan>
Dave Lewis (407) 242-4372
<dhl@ccd.harris.com>

GA – Atlanta:

Meets on the 1st Monday of each month in White Hall, Emory University.

Atlanta UNIX Users Group
P.O. Box 12241
Atlanta, GA 30355-2241
Mark Landry (404) 365-8108

KS or MO – Kansas:

Meets on 2nd Monday of each month.

Kansas City UNIX Users Group (KUUG)
813B Street
Blue Springs, MO 64015
(816) 235-5212
<mlg@cstp.umkc.edu>

MI – Detroit/Ann Arbor

Meets on the 2nd Thursday of each month in Ann Arbor.

Southeastern Michigan Sun Local Users Group
and Nameless UNIX Users Group
Steve Simmons office: (313) 769-4086
home: (313) 426-8981
<scs@lokkur.dexter.mi.us>

MN – Minneapolis/St. Paul:

Meets the 1st Wednesday of each month.

UNIX Users of Minnesota
17130 Jordan Court
Lakeville, MN 55044
Robert A. Monio (612) 220-2427
<pnessutt@dmsmq.mn.org>

MO – St. Louis:

St. Louis UNIX Users Group
P.O. Box 2182
St. Louis, MO 63158
Terry Linhardt (314) 772-4762
<uunet!jgaltstl!terry>

NE – Omaha:

Meets monthly.

/usr/group/nebraska
P.O. Box 31012
Omaha, NE 68132
Phillip Allendorfer (402) 423-1400

New England – Northern:

Meets monthly at different sites.

Peter Schmitt 603) 646-2085
Kiewit Computation Center
Dartmouth College
Hanover, NH 03755
<peter.schmitt@dartmouth.edu>

NJ – Princeton:

Meets monthly.

Princeton UNIX Users Group
Mercer County Community College
1200 Old Trenton Road
Trenton, NJ 08690
Peter J. Holsberg (609) 586-4800
<mccc!pjh>

NM – Albuquerque:

ASIGUNIX meets every 3rd Wednesday
of each month. Phil Hartz 505/275-0466.

NY – New York City:

Meets every other month in Manhattan.

Unigroup of New York City
G.P.O. Box 1931
New York, NY 10116
<unigroup@murphy.com>
Bob Young
(212) 490-8470

OK – Tulsa:

Meets 2nd Wednesday of each month.

Tulsa UNIX Users Group, \$USR
Stan Mason (918) 560-5329
<tulsix!smason@drd.com>

Mark Lawrence (918) 743-3013
<mark@drd.com>

TX – Austin:

Meets 3rd Thursday of each month.

Capital Area Central Texas UNIX Society
P.O. Box 9786
Austin, TX 78766-9786
<officers@cactus.org>
Tom Painter (512) 835-5457
<president@cactus.org>

TX – Dallas/Fort Worth:

Meets the 1st Thursday of each month.

Dallas/Fort Worth UNIX Users Group
P.O. Box 867405
Plano, TX 75086
Evan Brown (214) 519-3577
<evbrown@dsccc.com>

TX – Houston:

Meets 3rd Tuesday of each month.

Houston UNIX Users Group
(Hounix) answering machine (713) 684-6590
Bob Marcum, President (713) 270-8124
Chuck Bentley, Vice-president
(713) 789-8928
<chuckb@hounix.uucp>

WA – Seattle:

Meets monthly.

Seattle UNIX Group Membership Info.
Bill Campbell (206) 947-5591
6641 East Mercer
Mercer Island, WA 98040-0820
<bill@celestial.com>

CANADA – Manitoba:

Meets 2nd Tuesday of each month.

Manitoba UNIX User Group (MUUG)
P.O. Box 130, St. Boniface
Winnipeg, MB R2H 3B4
Bary Finch, President (204) 934-2723
<info@muug.mb.ca>

CANADA – Ottawa:

The Ottawa Carleton UNIX Users Group
D.J. Blackwood (613) 957-9305
<dave@revcan.rct.ca>

CANADA – Toronto:

143 Baronwood Court
Brampton, Ont. Canada L6V 3H8
Evan Leibovitch (416) 452-0504
<evan@telly.on.ca>

When your business depends on the Internet,
connect to the Internet service provider with
the most experience in the business.



A service of UUNET® Technologies, Inc.

A Full Spectrum of Services:

- Complete range of dedicated connections, all with no usage charges and unlimited e-mail, news, ftp, telnet, gopher, WWW
- Low-priced dial-up services with full IP access
- Nationwide points-of-presence
- Equipment configuration, domain name and IP address registration at no additional charge
- Free access to UUNET's extensive software archives
- On-site consulting services

Technical Excellence:

- Technical assistance from the Internet's top engineers, both during and after service installation
- Fully-staffed Network Operations Center, 24 hours a day/365 days a year

External Connectivity:

- Direct connections to more of the Internet's component networks worldwide than any other service provider
- Founding member of the CIX and MAE-East

Network Superiority:

- The Internet's first 45 Mbps ATM backbone
- Customer access up to 10 Mbps
- More redundant connections within our network than any other Internet service provider, minimizing customer exposure to network outages



For more information,
contact an AlterNet
representative.

Voice: +1 800 258 4037

E-mail: alternet-info@alter.net

Serving your Internet needs since 1987

Calendar of Events

1994

- Apr 4-8 * SANS III, Arlington, VA
11-14 * C++, Cambridge, MA
11-15 EurOpen Forum, Egham, Surrey, UK
18-22 IEEE 1003, Lake Tahoe, CA
19-21 SUUG: Open Systems: Solutions for an Open World, Moscow, Russia
25-28 * UNIX Applications Development Toronto, Canada
- May 2-6 NetWorld+INTEROP 94, Las Vegas, NV
7-13 DECUS, New Orleans, LA
- Jun 6-10 * USENIX, Boston, MA
" NetWorld+INTEROP 94, Berlin
16-18 SUG, San Francisco, CA
- Jul 11-15 IEEE 1003
- Aug 1-2 * High-Speed Networking, Berkeley, CA
- Sept 6-9 AUUG, Melbourne, Australia
12-14 NetWorld+INTEROP 94, Atlanta, GA
18-22 Interex 94, Denver, CO
20-22 GUUG, Wiesbaden, Germany
19-23 * LISA VIII, San Diego, CA
- Oct 17-21 IEEE 1003
23-27 ACM OOPSLA, Portland, OR
26-28 * Very High Level Languages, Santa Fe, NM
- Nov 12-18 DECUS, Anaheim, CA
14-18 * OSDI, Monterey, CA
" SUG Technical Workshop, Austin, TX
- Dec 8-9 * Mobile Computing Systems and Applications, Santa Cruz, CA

1995

- Jan 16-20 * USENIX, New Orleans, LA
- Feb 21-23 UniForum, Dallas, TX
- May 13-19 DECUS, Atlanta, GA
- Aug 13-17 Interex 95, Toronto, Canada
- Nov 2-8 DECUS, San Francisco, CA

1996

- Jan 22-26 * USENIX, San Diego, CA
- Mar 12-14 UniForum, San Francisco, CA
- May 18-24 DECUS, Orlando, FL
- June 17-21* Washington, DC
- Aug 4-8 Interex 96, San Diego, CA
- Nov 16-22 DECUS, Anaheim, CA

This is a combined calendar of planned conferences, symposia, and standards meetings related to the UNIX operating system. If you have a UNIX-related event that you wish to publicize, please contact login@usenix.org. Please provide your information in the same format as above.

* = events sponsored by the USENIX Association.

ACM: Association for Computing Machinery
AUUG: Australian UNIX Users Group
DECUS: Digital Equipment Computer Users Society
EurOpen: European Forum for Open Systems

FedUNIX: Council of Advanced Computing Systems Technologists in Government
GURU: Roumanian UNIX User Group
GUUG: German UNIX Users Groups

IEEE: Institute of Electrical and Electronics Engineers
IETF: Internet Engineering Task Force
INET: Internet Society
Interex: Intl. Association of Hewlett-Packard Comp.Users

JUS: Japan UNIX Society
LISA: USENIX Systems Administration Conference
NOSSDAV: Network and Operating System Support for Digital Audio and Video

OOPSLA: Object - oriented Programming Systems, Languages, and Applications

OSDI: Symposium on Operating Systems Design & Implementation
SAGE: System Administrators' Guild
SANS: System Administration, Networking & Security
SUG: Sun User Group
SUUG: Soviet UNIX Users Group

UKUUG: United Kingdom UNIX Systems Users Group
UniForum: International Association of UNIX and Open Systems Professionals

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710

Second Class Postage
PAID
At Berkeley, California
and Additional Offices

POSTMASTER: Send address changes to ;login:, USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710



UPCOMING SYMPOSIA AND CONFERENCES



APRIL 4-8, 1994

**1994 WORLD CONFERENCE ON
TOOLS AND TECHNIQUES FOR
SYSTEM ADMINISTRATION,
NETWORKING, AND SECURITY
(SANS-III)**

Stouffers Concourse Hotel
Arlington, Virginia

Sponsored by The Open Systems Conference Board
in cooperation with the USENIX Association,
SAGE, the System Administrators Guild, and others

APRIL 11-14, 1994

**6TH C++ CONFERENCE
& ADVANCED TOPICS
WORKSHOP**

Marriott Hotel
Cambridge, Massachusetts
Program Chair: Doug Lea
*State University of New York at
Oswego*



APRIL 25-28, 1994

**UNIX APPLICATIONS
DEVELOPMENT SYMPOSIUM**

Marriott Hotel
Toronto, Ontario, Canada
Program Chair: Jim Duncan
Pennsylvania State University
Program Vice Chair: Greg Woods
GAW Consulting



JUNE 6-10, 1994

**SUMMER 1994
TECHNICAL CONFERENCE**

Marriott Hotel
Boston, Massachusetts
Program Chairs: Margo Seltzer
Harvard University, and Keith Bostic
University of California, Berkeley



AUGUST 1-2, 1994

**SYMPOSIUM ON
HIGH-SPEED NETWORKING
with field trips on August 3**

Claremont Hotel & Resort
Oakland, California
Program Chair: Pat Parseghian,
AT&T Bell Laboratories



SEPTEMBER 19-23, 1994

**8TH SYSTEM ADMINISTRATION
CONFERENCE (LISA VIII)**

Town & Country Hotel
San Diego, California
Program Chair: Dinah McNutt,
Zilker Internet Park, Inc.
Co-sponsored with SAGE,
the System Administrators Guild



TO RECEIVE FULL INFORMATION

Please contact: **USENIX Conference Office**, 22672 Lambert St., Suite 613, Lake Forest, CA USA 92630
+1 (714) 588-8649; FAX: +1 (714) 588-9706; e-mail: conference@usenix.org